RF Pulse Design: Multi-Dimensional Excitation

M229 Advanced Topics in MRI

Holden H. Wu, Ph.D.

2025.04.17

Department of Radiological Sciences
David Geffen School of Medicine at UCLA

Class Business

- Office hours
  - Holden: by appointment
  - Wenqi (HW1): 10-12 on 4/18 Fri
  - Timo (HW2): 4/18, 4/24, 4/25
  - Email beforehand
- Homework 1 due on 4/21 Mon
- Homework 2 due on 4/28 Mon
- Final project
  - Start thinking

Outline

- Review of adiabatic pulses
- Small tip approximation
- Excitation k-space interpretation
- 2D EPI pulse design
- MATLAB demo
- Homework 2

Review of Adiabatic Pulses

Adiabatic Pulses

- Flip Angle $\neq \int_0^T B_1(t)dt$
- Amplitude and frequency modulation
- Long duration (8-12 ms)
- High B_1 amplitude (>12 μT)
- Generally NOT multi-purpose (inversion pulses cannot be used for refocusing, etc.)

Non-adiabatic Pulses

- Flip Angle $= \int_0^T B_1(t)dt$
- Amplitude modulation with constant carrier frequency
- Short duration (0.3-1 ms)
- Low B_1 amplitude
- Generally multi-purpose (inversion pulses can be used for refocusing, etc.)

Bloch Equation

$$\frac{d\vec{M}}{dt} = \vec{M} \times \gamma \vec{B}_{eff}$$

Non-selective vs. Selective Excitation

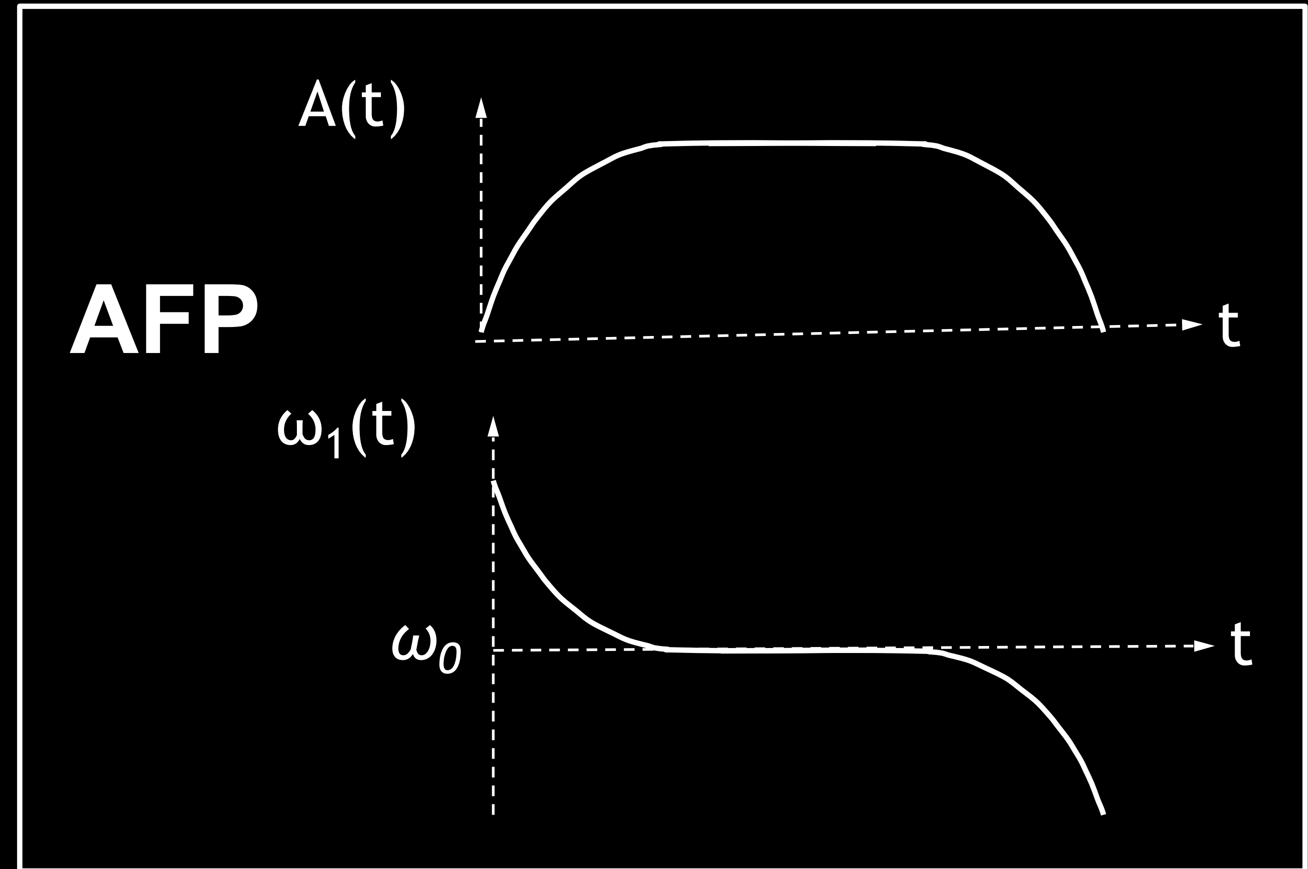
$$\vec{B}_{eff} = \begin{pmatrix} B_1(t) \\ 0 \\ B_0 - \frac{\omega}{\gamma} \end{pmatrix}$$

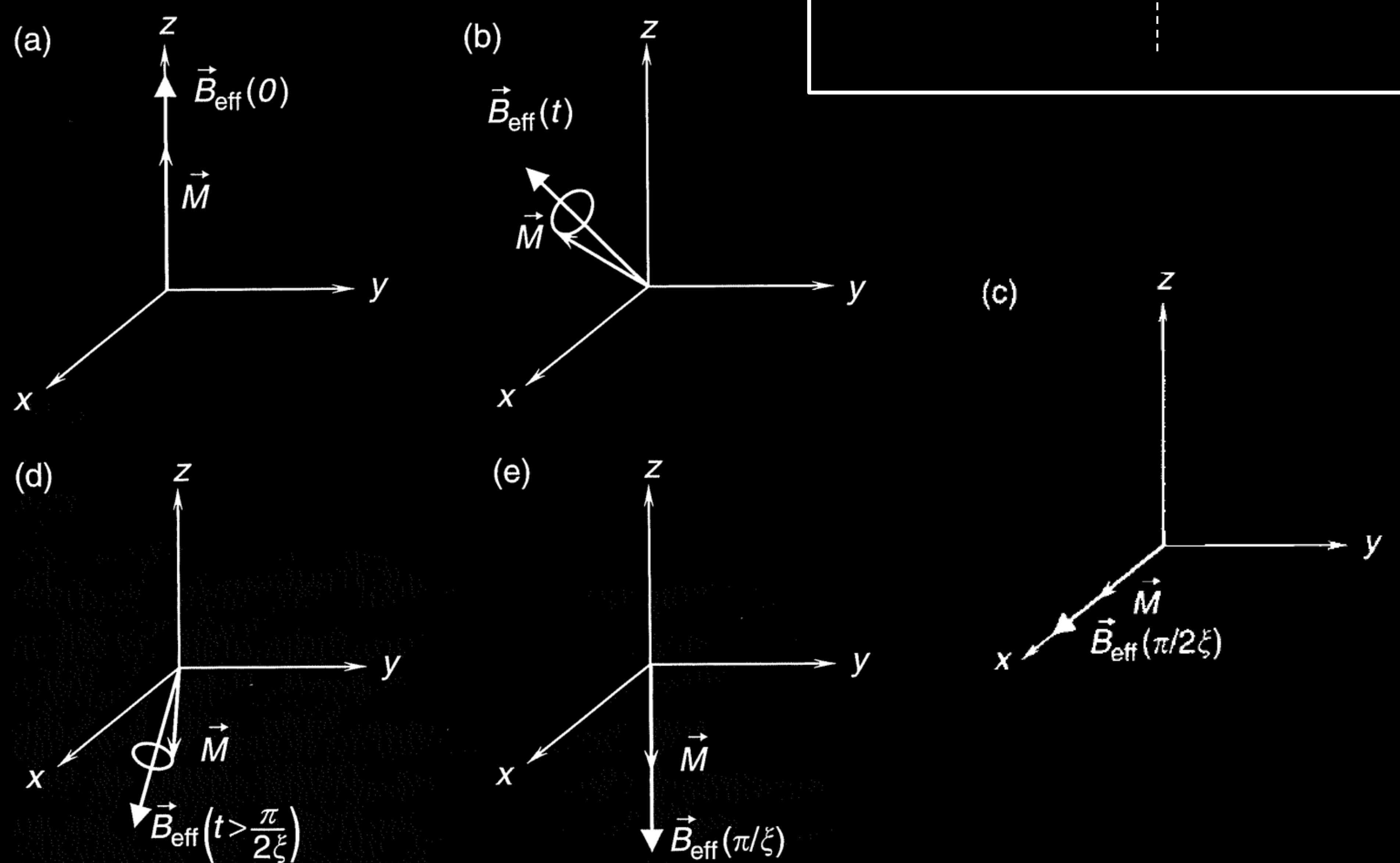
$$\vec{B}_{eff} = \begin{pmatrix} B_1(t) \\ 0 \\ B_0 - \frac{\omega}{\gamma} + G_z z \end{pmatrix}$$

Adiabatic Pulses

$$\vec{B}_{eff} = \begin{pmatrix} A(t) \\ 0 \\ B_0 - \frac{\omega}{\gamma} + \frac{\omega_1(t)}{\gamma} \end{pmatrix}$$

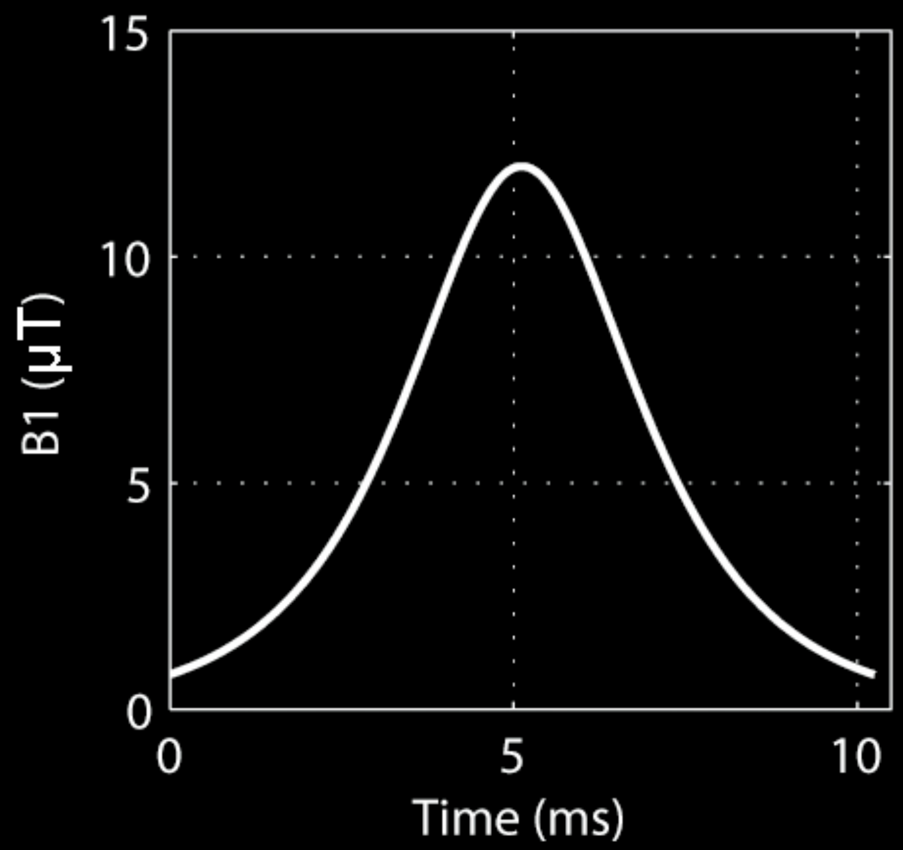
Adiabatic Inversion



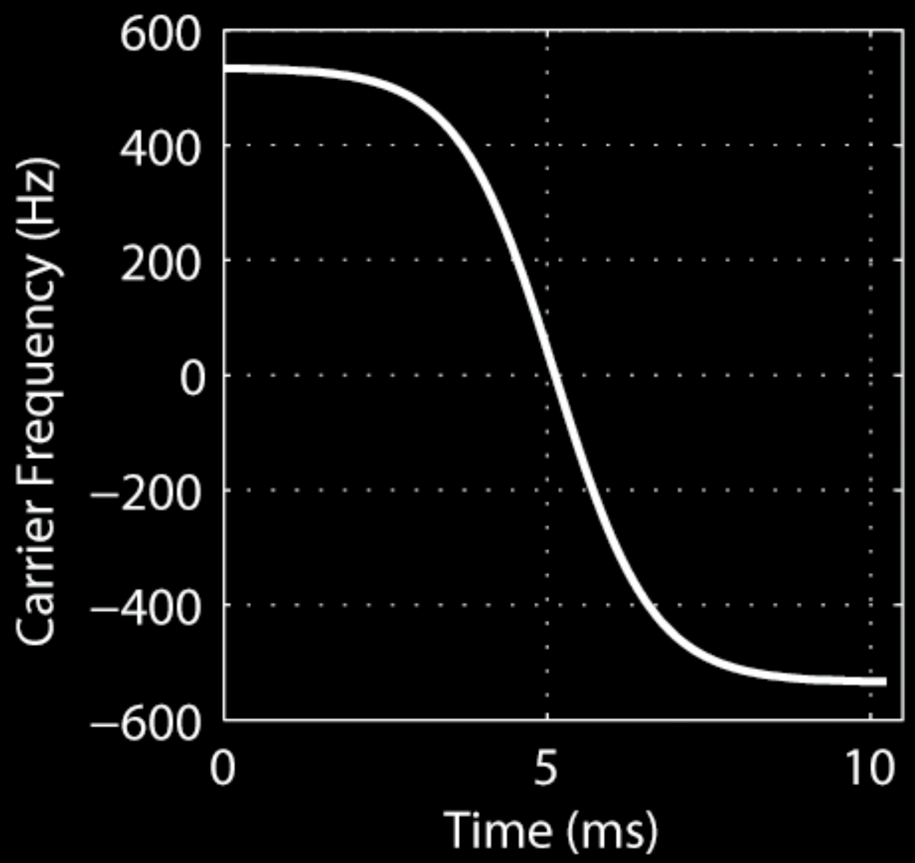


Hyperbolic Secant Pulse Example

Amplitude Modulation, A(t)



Frequency Modulation, $\omega_1(t)$



Pulse Parameters:

A_0 = 12 μT

μ = 5

β = 672 rad/s

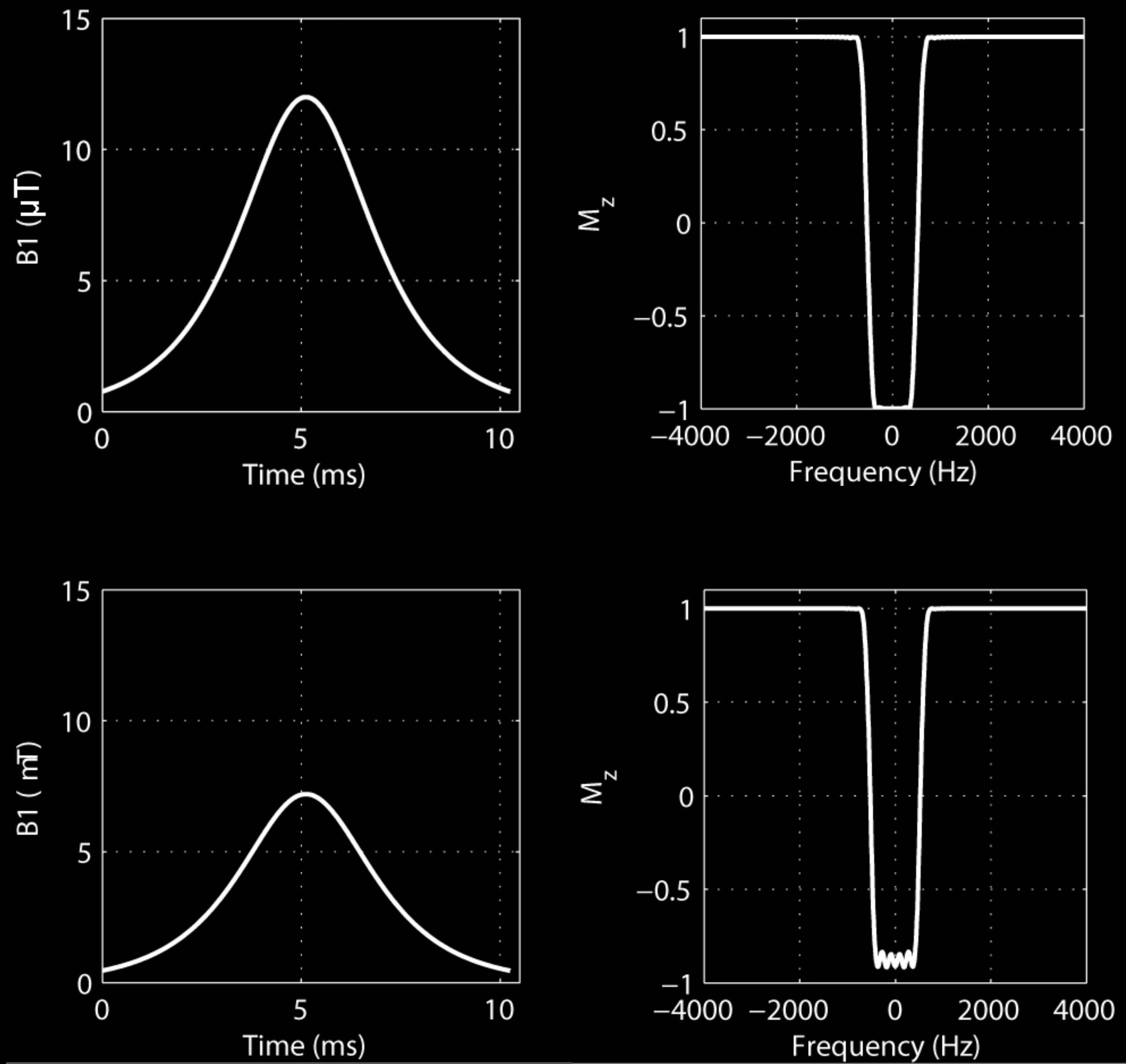
Duration = 10.24 ms

Hyperbolic Secant: Adiabatic Property

Original Pulse (100%)
B_{1max} = 12 µT

60% Attenuated Pulse
B_{1max} = 7.2 µT

B_1 Threshold ≈ 6 µT



Small Tip Approximation

Bloch Equation (at on-resonance)

$$\frac{d\vec{M}_{rot}}{dt} = \vec{M}_{rot} \times \gamma \vec{B}_{eff}$$

where $\vec{B}_{eff} = \begin{pmatrix} B_1(t) \\ 0 \\ \cancel{B_0 - \frac{\omega}{\gamma}} + G_z z \end{pmatrix}$

When we simplify the cross product,

$$\frac{d\vec{M}}{dt} = \begin{pmatrix} 0 & \omega(z) & 0 \\ -\omega(z) & 0 & \omega_1(t) \\ 0 & -\omega_1(t) & 0 \end{pmatrix} \vec{M}$$

$$\omega(z) = \gamma G_z z \qquad \omega_1(t) = \gamma B_1(t)$$

Small Tip Approximation

$$\frac{d\vec{M}}{dt} = \begin{pmatrix} 0 & \omega(z) & 0 \\ -\omega(z) & 0 & \omega_1(t) \\ 0 & -\omega_1(t) & 0 \end{pmatrix} \vec{M}$$

$M_z \approx M_0$ small tip-angle approximation

sin θ ≈ θ
cos θ ≈ 1
$M_z \approx M_0 \rightarrow$ constant

$$\frac{dM_z}{dt} = 0$$

$$\frac{dM_{xy}}{dt} = -i\gamma G_z z M_{xy} + i\gamma B_1(t) M_0$$

$$M_{xy} = M_x + iM_y$$

First order linear differential equation. Easily solved.

$$\frac{dM_{xy}}{dt} = -i\gamma G_z z M_{xy} + i\gamma B_1(t) M_0$$

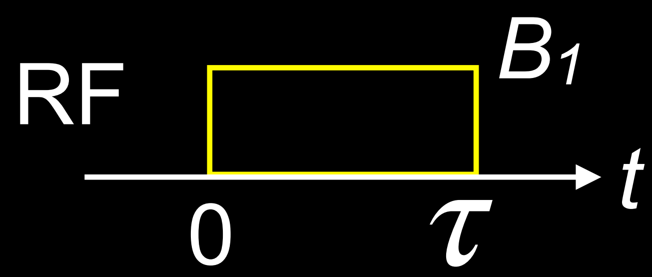
Solving a first order linear differential equation:

$$M_{xy}(t,z) = i\gamma M_0 \int_0^t B_1(s) e^{-i\gamma G_z z\cdot(t-s)} ds$$

$$M_r(\tau, z) = iM_0 e^{-i\omega(z)\tau/2} \cdot \mathcal{FT}_{1D}\{\omega_1(t + \frac{\tau}{2})\} \,|_{f=-(\gamma/2\pi)G_z z}$$

(See the references for complete derivation.)

$$M_r(\tau, z) = iM_0 e^{-i\omega(z)\tau/2} \cdot FT_{1D}\{\omega_1(t+\frac{\tau}{2})\}\,|_{f=-(\gamma/2\pi)G_z z}$$



Simple RF pulse: $B_1(t) = B_1 \cdot \sqcap(\frac{t-\frac{\tau}{2}}{\tau})$

$$\omega_1(t+\frac{\tau}{2}) = \gamma \cdot B_1 \cdot \sqcap(\frac{t}{\tau})$$

$$FT_{1D}\{\sqcap(\frac{t}{\tau})\} = \tau \cdot sinc(\tau \cdot f) = \tau \cdot sinc(\tau \cdot \frac{\gamma}{2\pi} G_z \cdot z)$$

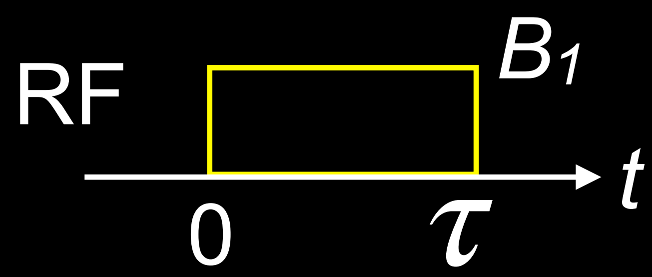
$$\Rightarrow M_{xy}(z) = iM_0 e^{-i\omega(z)\cdot\tau/2} \cdot \omega_1 \cdot \tau \cdot sinc(\tau \cdot \frac{\gamma}{2\pi} G_z \cdot z)$$

$$M_{xy}(z) \propto sinc(\tau \cdot \frac{\gamma}{2\pi} G_z \cdot z)$$

Note: τ is at the end of the RF pulse

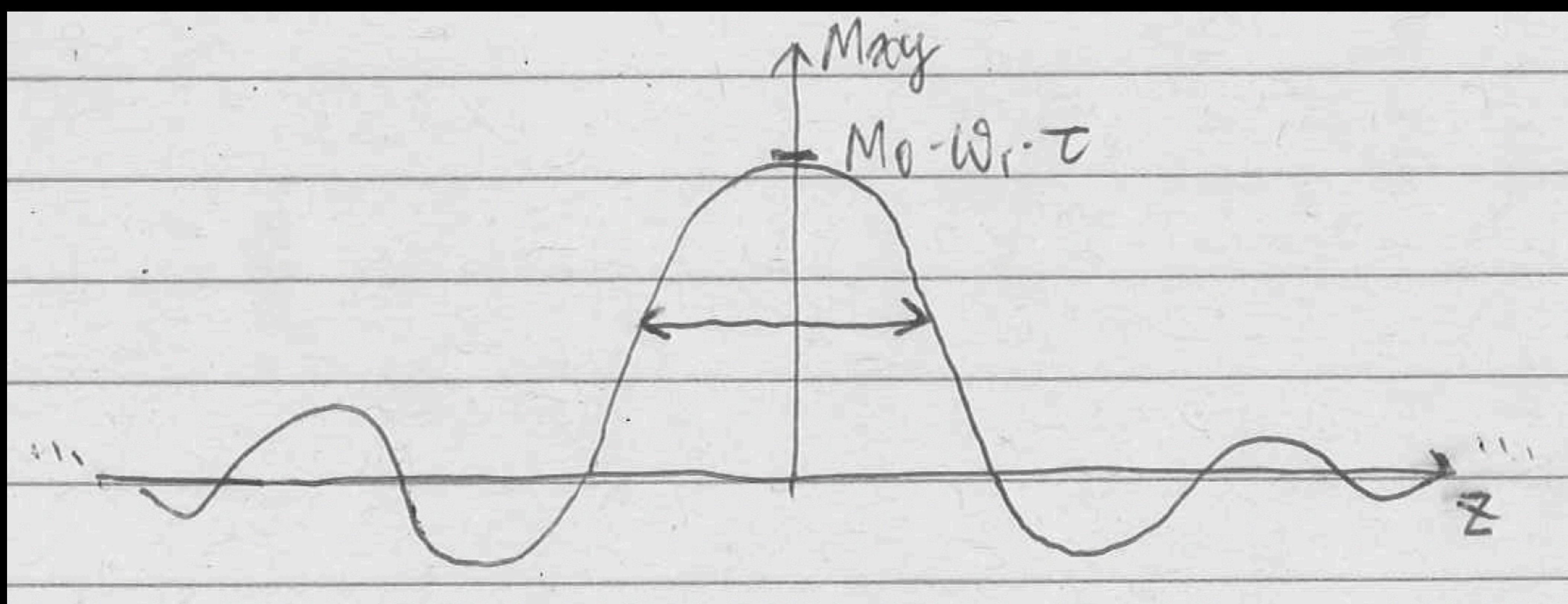
Extra phase can be rephased using a G_z gradient

$$M_r(\tau, z) = iM_0 e^{-i\omega(z)\tau/2} \cdot FT_{1D}\{\omega_1(t + \frac{\tau}{2})\} \,|_{f=-(\gamma/2\pi)G_z z}$$



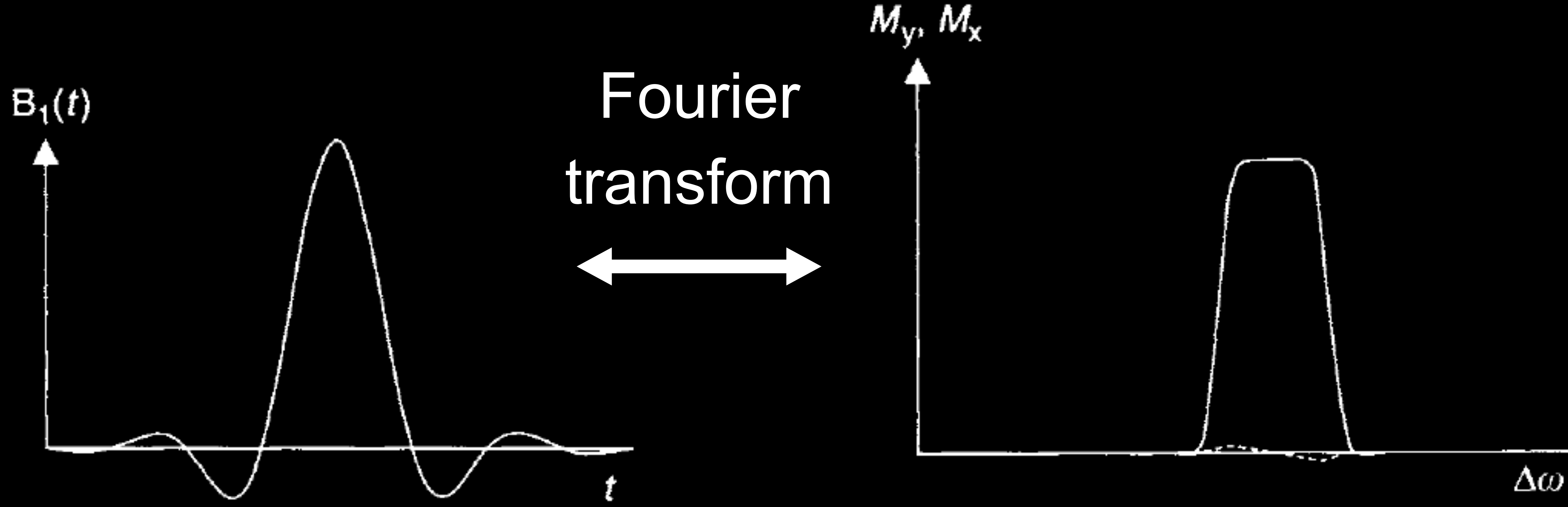
Simple RF pulse: $B_1(t) = B_1 \cdot \Pi\left(\frac{t - \frac{\tau}{2}}{\tau}\right)$

$$M_{xy}(z) \propto sinc(\tau \cdot \frac{\gamma}{2\pi} G_z \cdot z)$$



What if we want to excite a rectangular slice?

Small Tip Approximation



- For small tip angles, “the slice or frequency profile is well approximated by the Fourier transform of $B_1(t)$”
- The approximation works surprisingly well even for flip angles up to 90°

Excitation k-space Interpretation

Small Tip Approximation

$$M_{xy}(t,z) = i\gamma M_0 \int_0^t B_1(s) e^{-i\omega(z)(t-s)} ds$$

$$\omega(z) = \gamma G_z z \;\Rightarrow\; \omega(\vec{r},t) = \gamma \vec{G}(t)\vec{r}$$

$$M_{xy}(t,\vec{r}) = i\gamma M_0 \int_0^t B_1(s) e^{-i\gamma \int_s^t \vec{G}(\tau) d\tau \cdot \vec{r}} ds$$

Small Tip Approximation

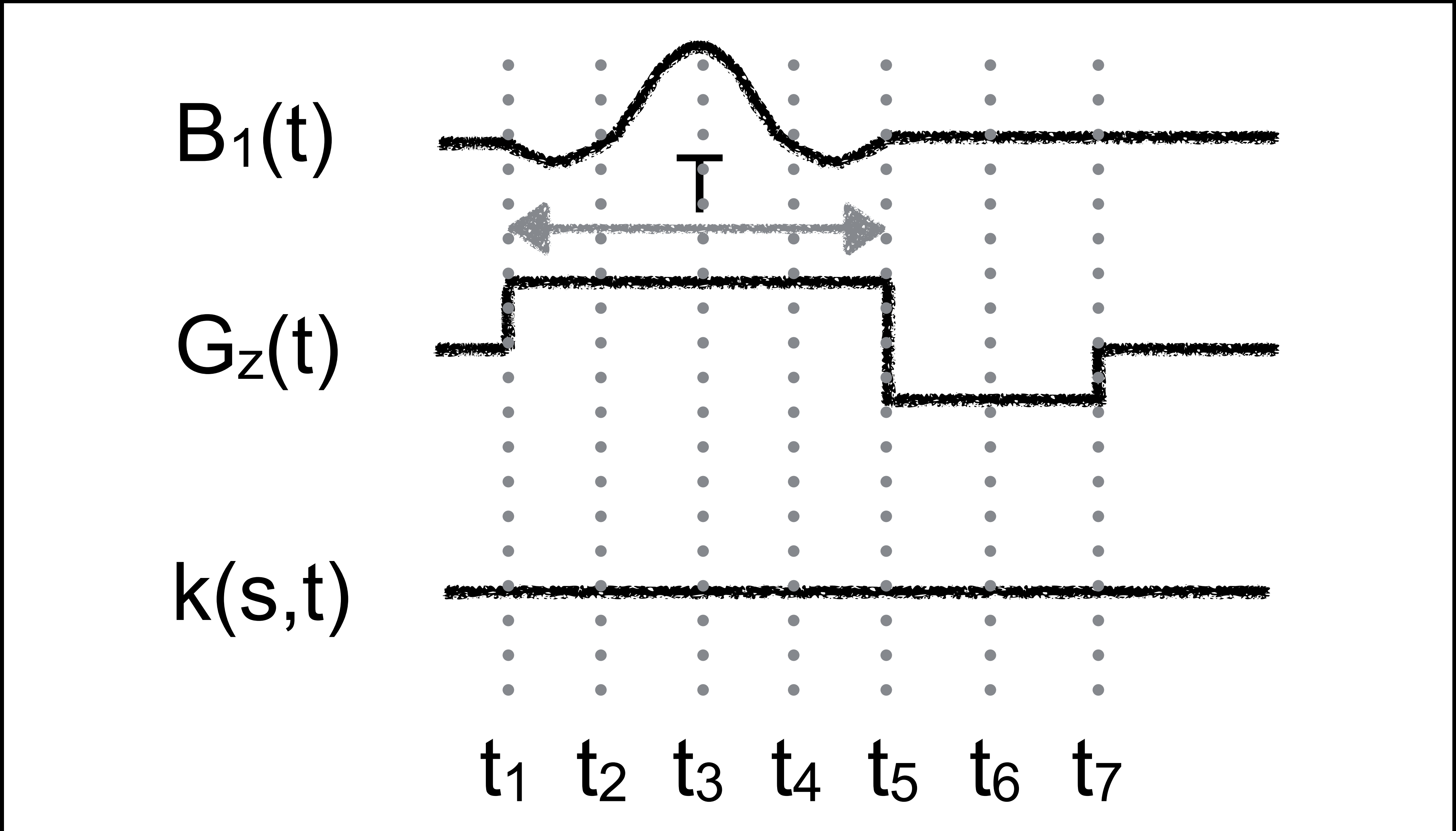
$$M_{xy}(t,\vec{r}) = i\gamma M_0 \int_0^t B_1(s) e^{-i\gamma \int_s^t \vec{G}(\tau)d\tau \cdot \vec{r}} ds$$

Let us define: $\vec{k}(s,t) = -\frac{\gamma}{2\pi} \int_s^t \vec{G}(\tau)d\tau$

$$M_{xy}(t,\vec{r}) = i\gamma M_0 \int_0^t B_1(s) e^{i2\pi \vec{k}(s,t)\cdot \vec{r}} ds$$

One-Dimensional Example

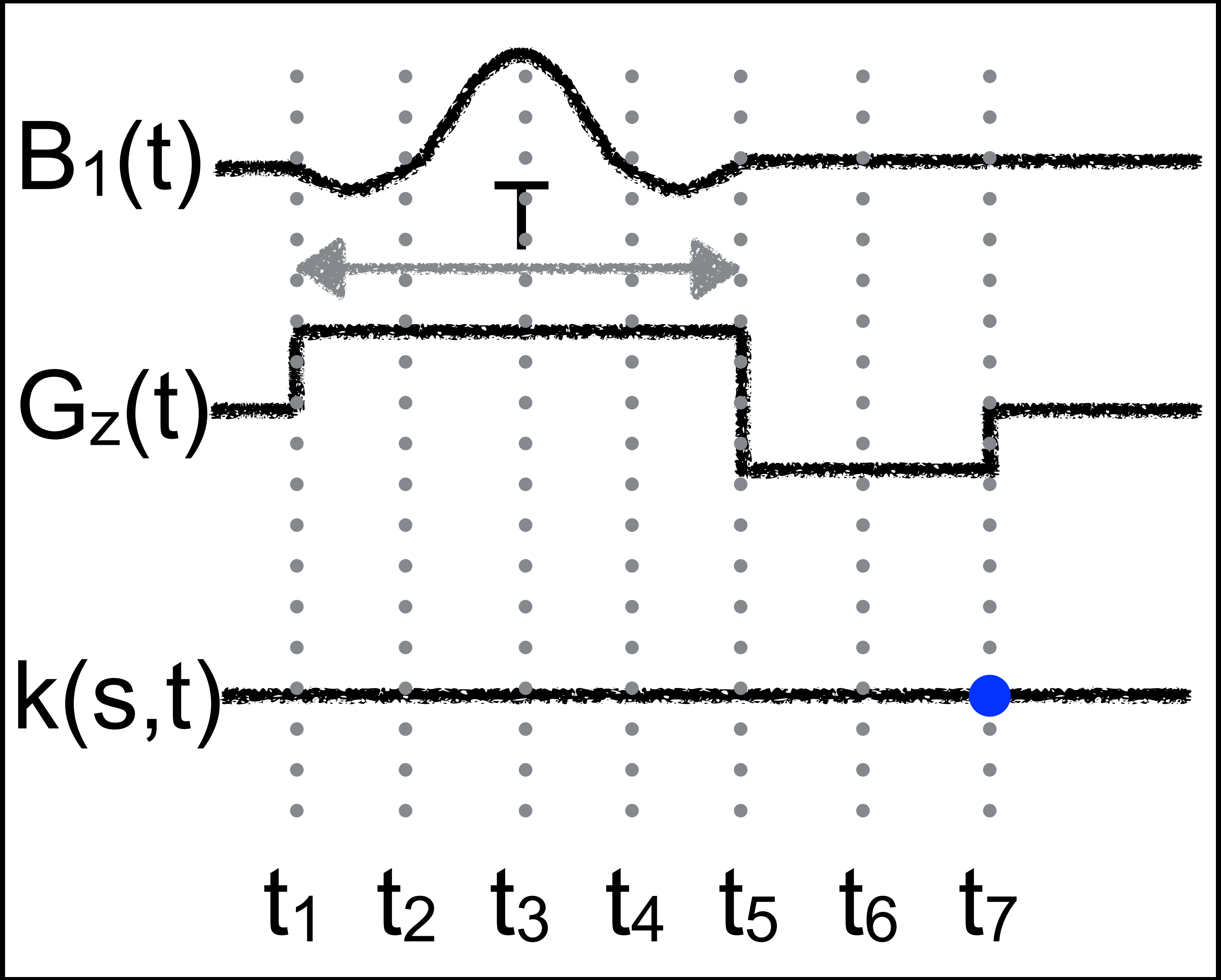
$$\vec{k}(s,t) = -\frac{\gamma}{2\pi}\int_s^t \vec{G}(\tau)d\tau$$



Consider the value of **k** at s = t_1, t_2, … t_7

One-Dimensional Example

$$\vec{k}(s,t) = -\frac{\gamma}{2\pi}\int_s^t \vec{G}(\tau)d\tau$$



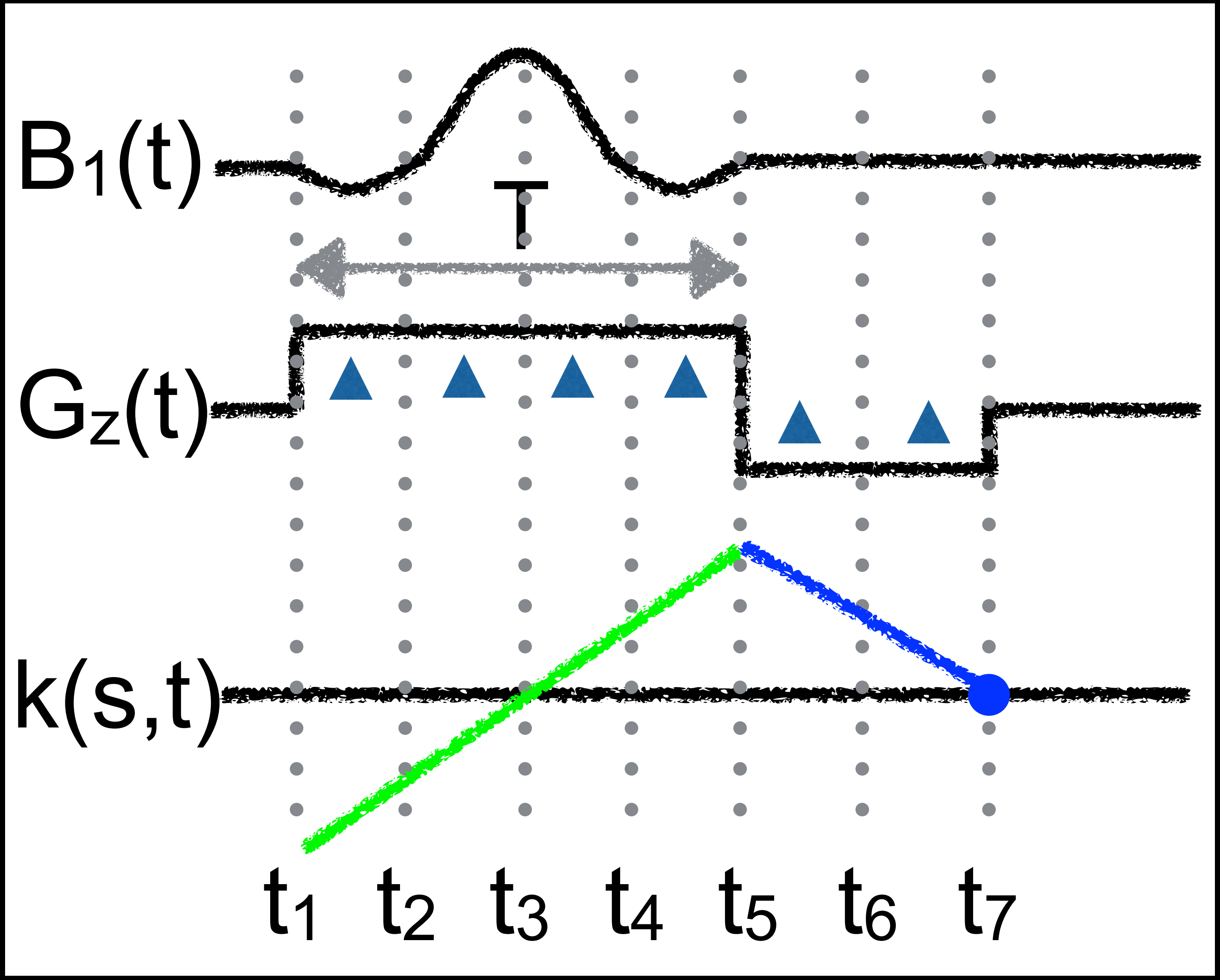
At s = t_7

$$\vec{k}(s) = -\frac{\gamma}{2\pi}\int_{t_7}^{t_7} \vec{G}(\tau)d\tau = 0$$

End of pulse is t = t_7

One-Dimensional Example

$$\vec{k}(s,t) = -\frac{\gamma}{2\pi}\int_s^t \vec{G}(\tau)d\tau$$



At s = t_7, $\vec{k}(s) = 0$

At s = t_6, $\vec{k}(s) = +\Delta$

At s = t_5, $\vec{k}(s) = +2\Delta$

At s = t_4, $\vec{k}(s) = +\Delta$

At s = t_3, $\vec{k}(s) = 0$

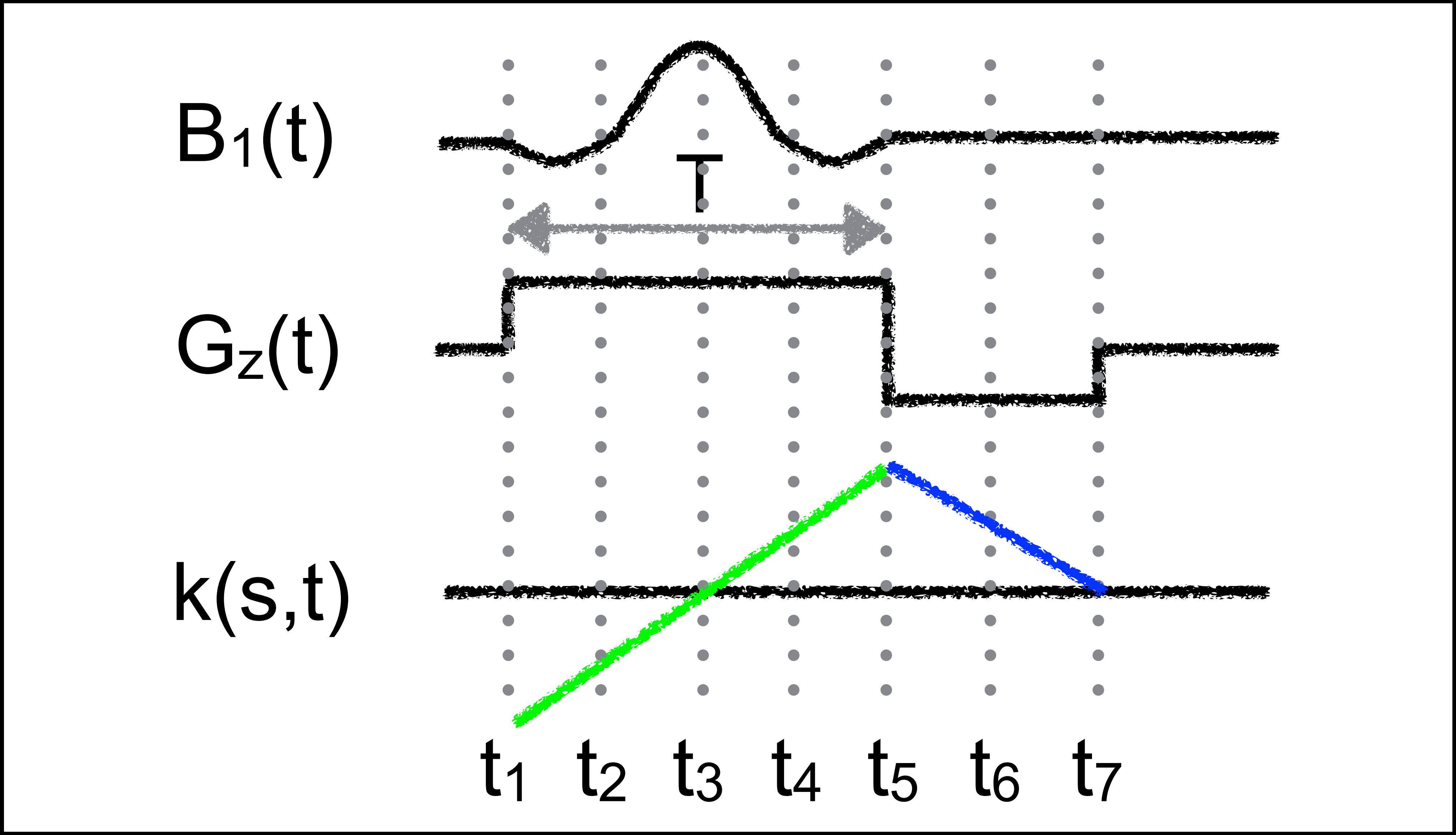
At s = t_2, $\vec{k}(s) = -\Delta$

At s = t_1, $\vec{k}(s) = -2\Delta$

End of pulse is t = t_7

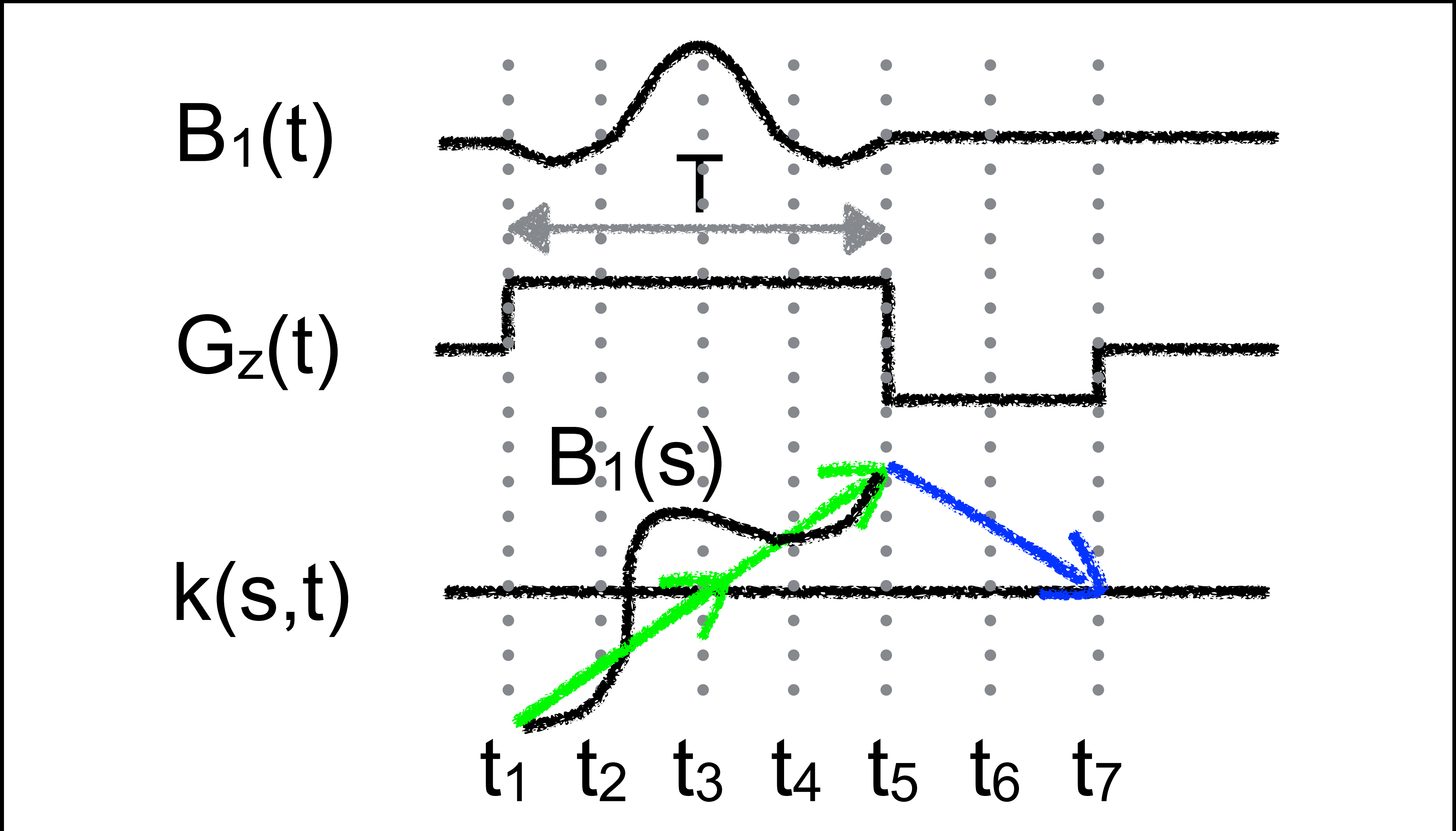
One-Dimensional Example

$$\vec{k}(s,t) = -\frac{\gamma}{2\pi}\int_s^t \vec{G}(\tau)d\tau$$

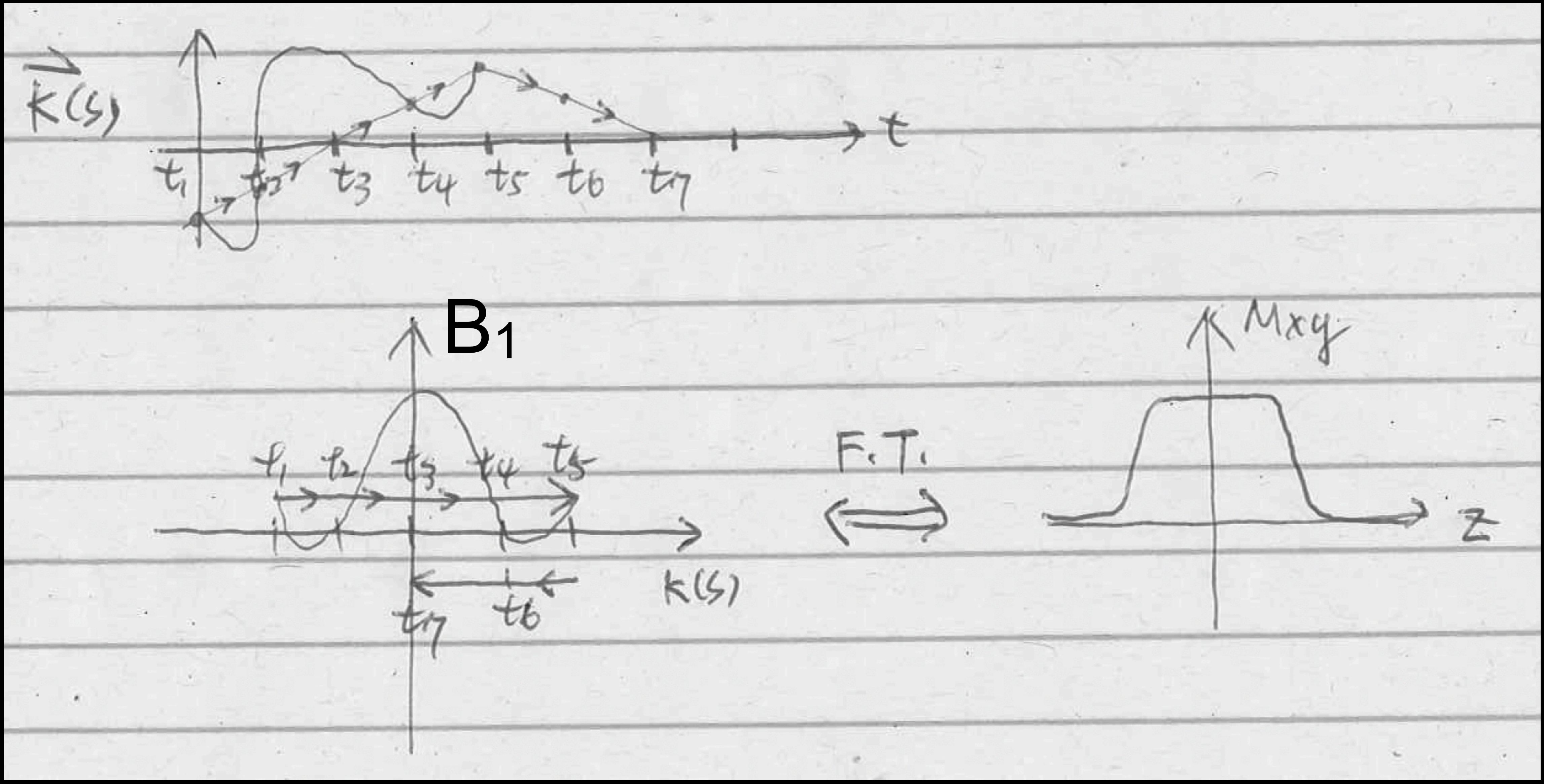


One-Dimensional Example

$$\vec{k}(s,t) = -\frac{\gamma}{2\pi}\int_s^t \vec{G}(\tau)d\tau$$



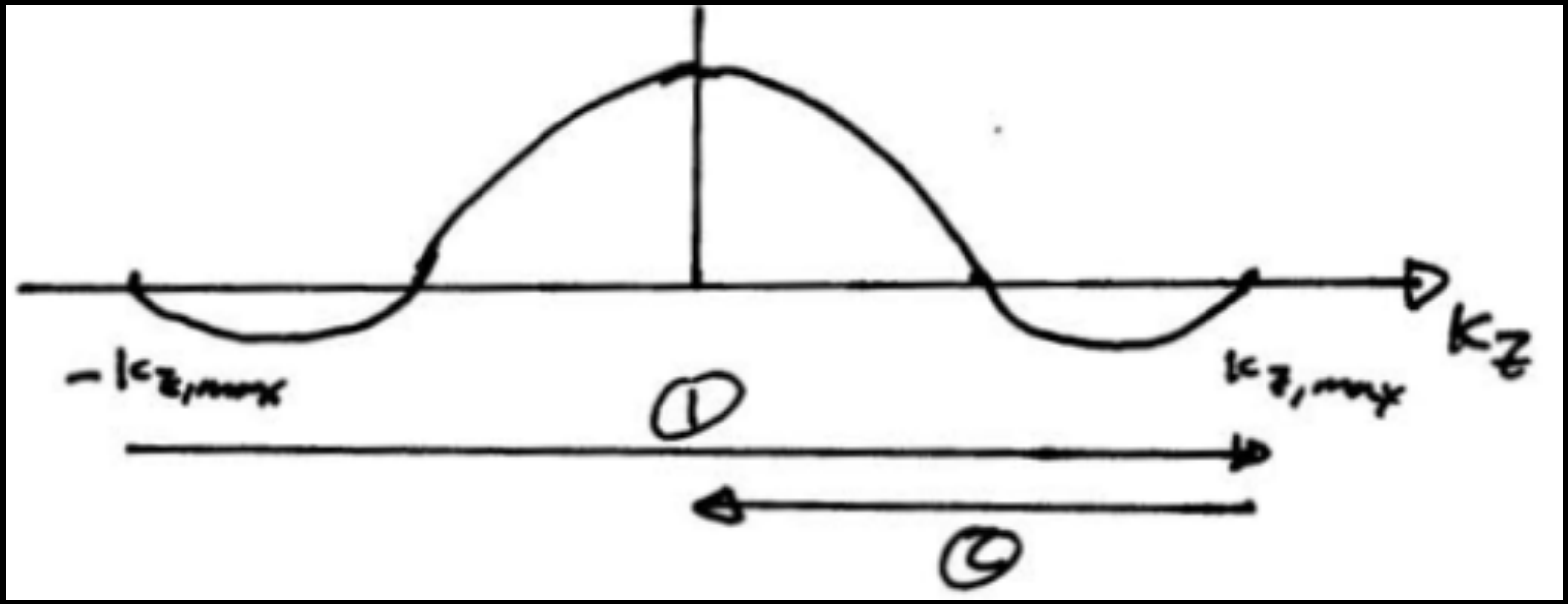
One-Dimensional Example



Excitation k-space

Excited slice profile

One-Dimensional Example



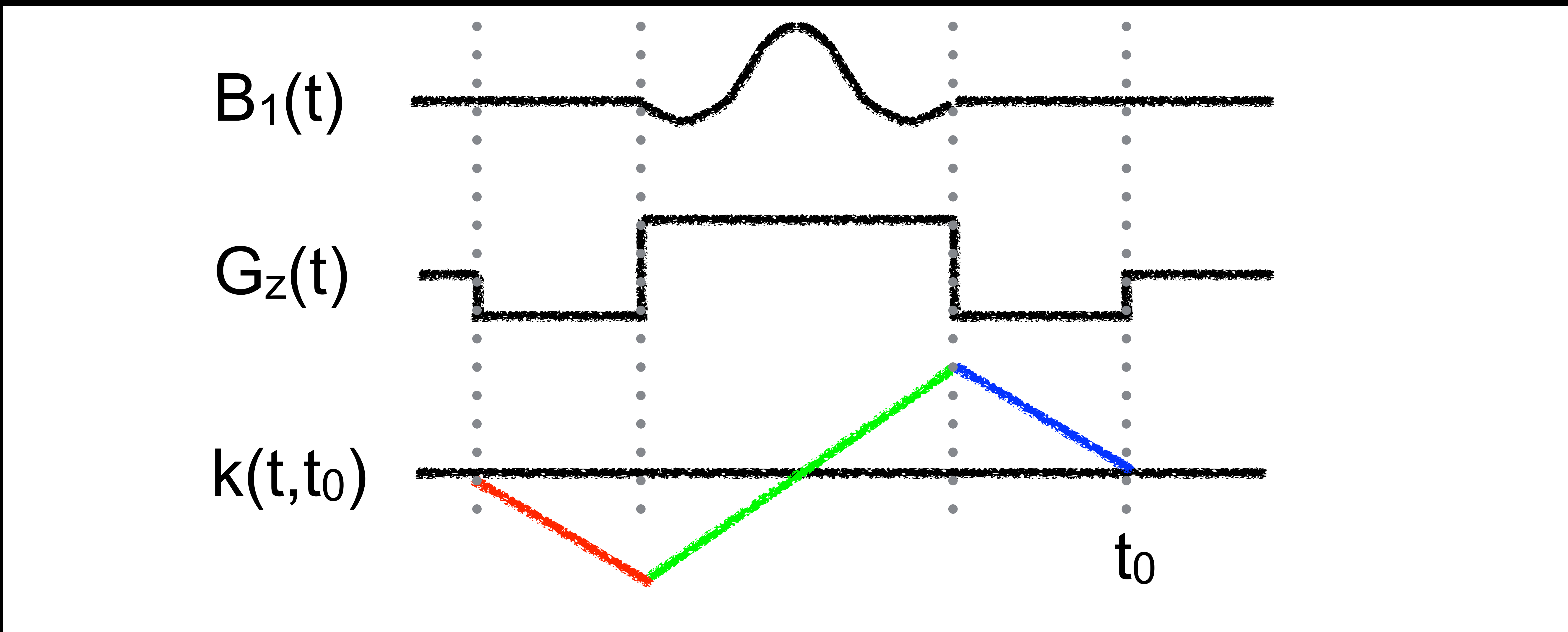
$$k_{z,max} = \frac{T}{2}\frac{\gamma}{2\pi}G_z$$

- This gives magnetization at t = t_0, the end of the pulse
- Looks like you scan across k-space, then return to origin

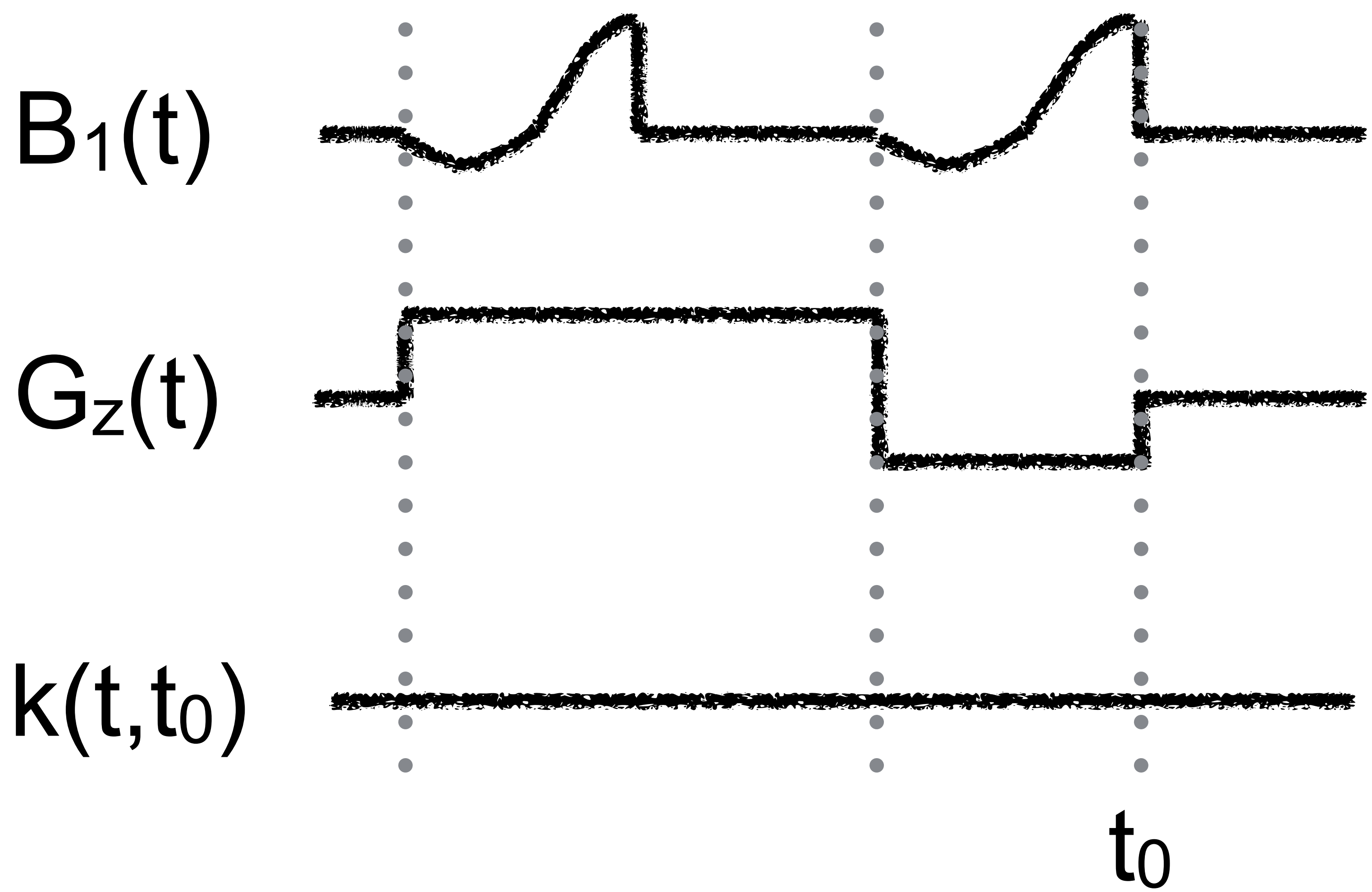
Evolution of Magnetization During Pulse

- RF pulse goes in at DC ($k_z = 0$)
- Gradients move previously applied weighting around
- Think of the RF as "writing" an analog waveform in k-space
- The effect of rephasing gradients
- Same idea applies to reception

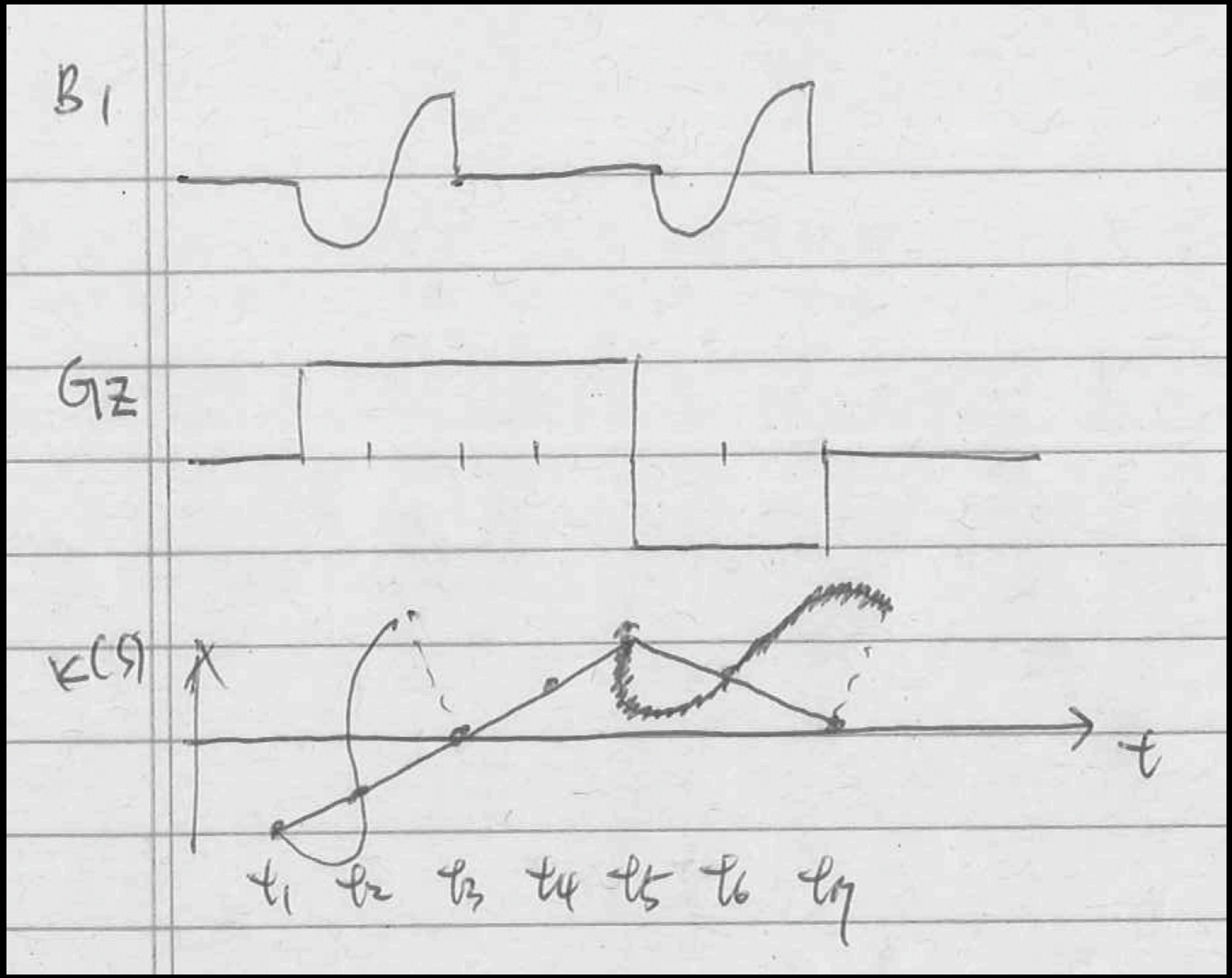
Other 1D Examples

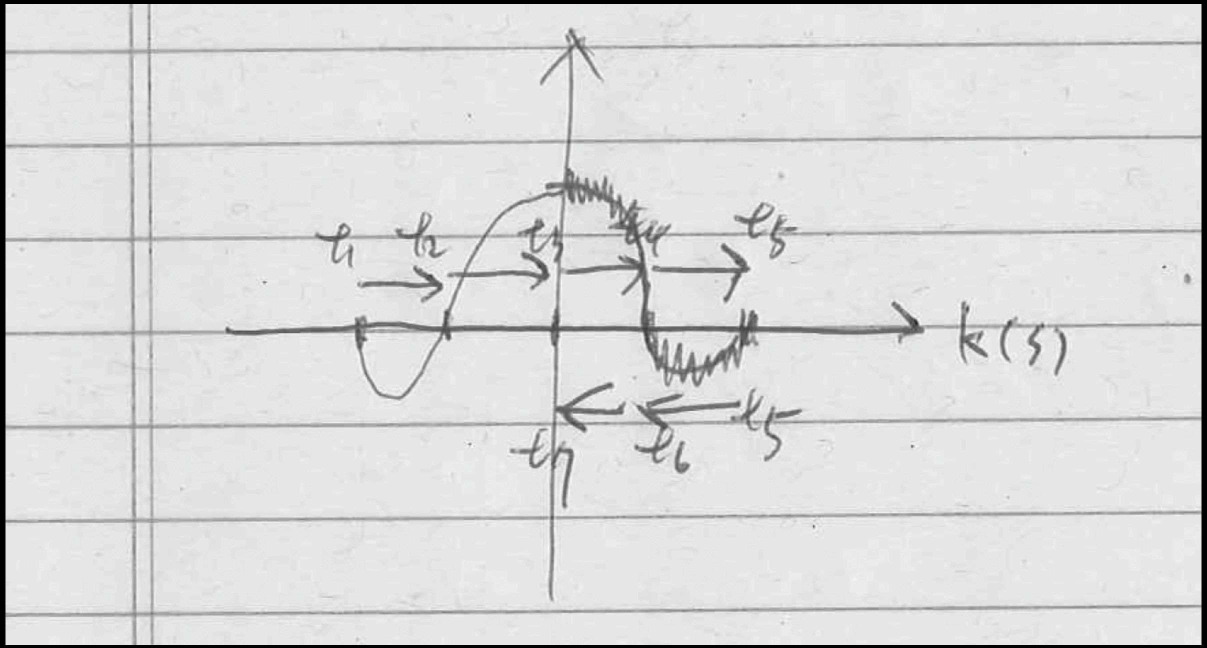


Other 1D Examples

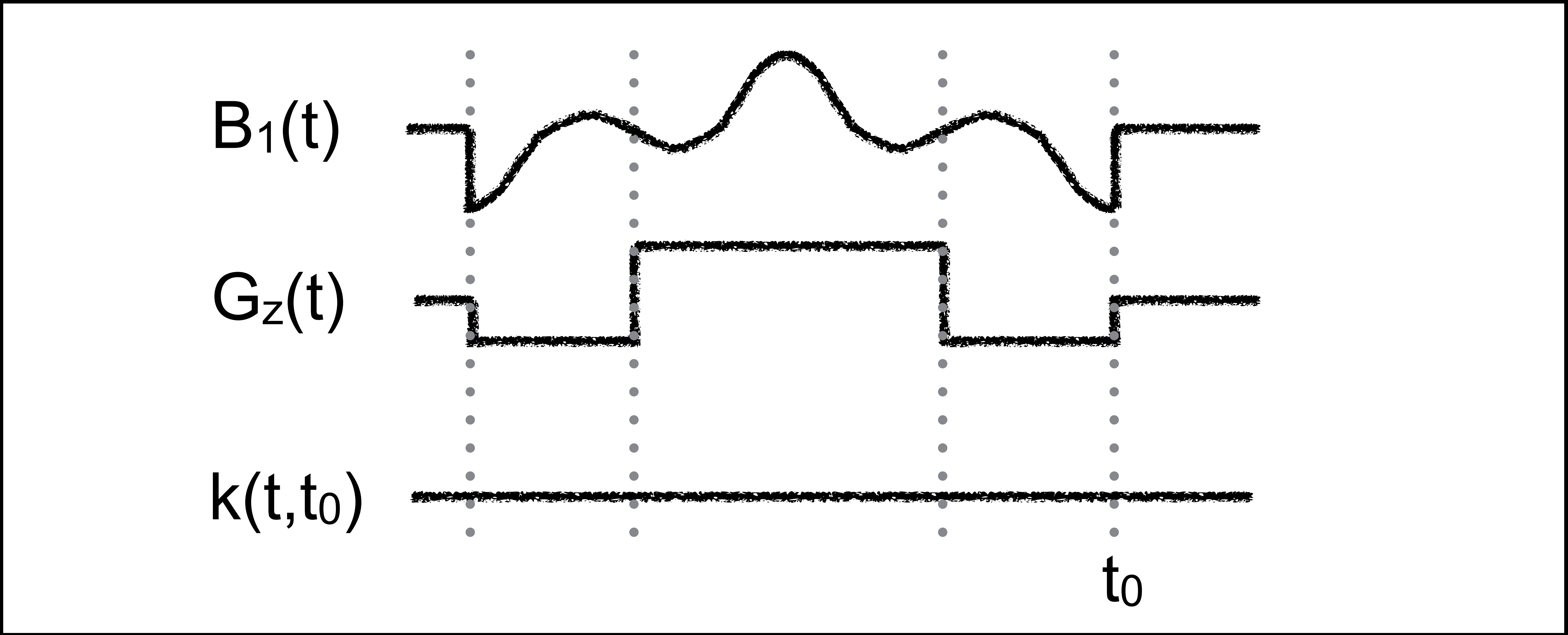


Other 1D Examples





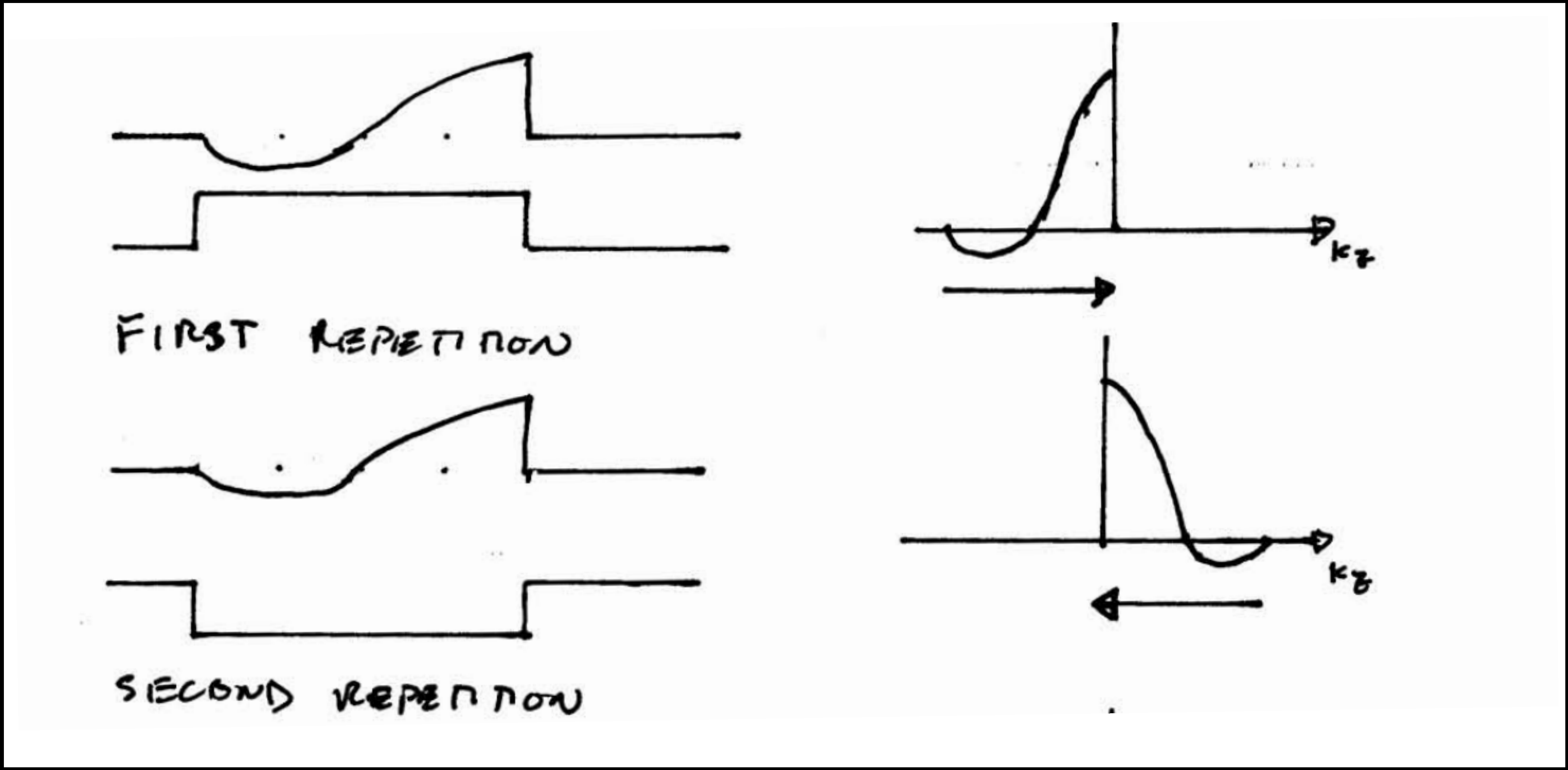
Other 1D Examples



Multiple Excitations

- Most acquisition methods require several repetitions to make an image
  - e.g., 128 phase encodes
- Data is combined to reconstruct an image
- Same idea works for excitation!
  - Build up the excitation profile by traversing excitation k-space and depositing RF energy

Simple 1D Example



Sum the data from two acquisitions

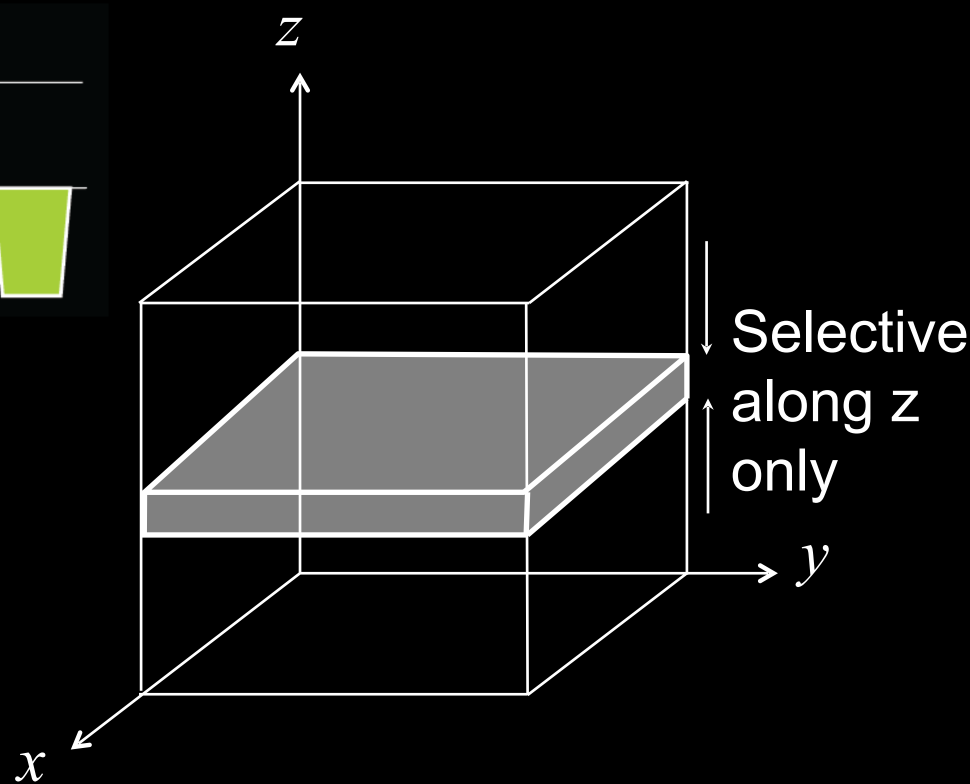
Same profile as slice selective pulse, but zero echo time

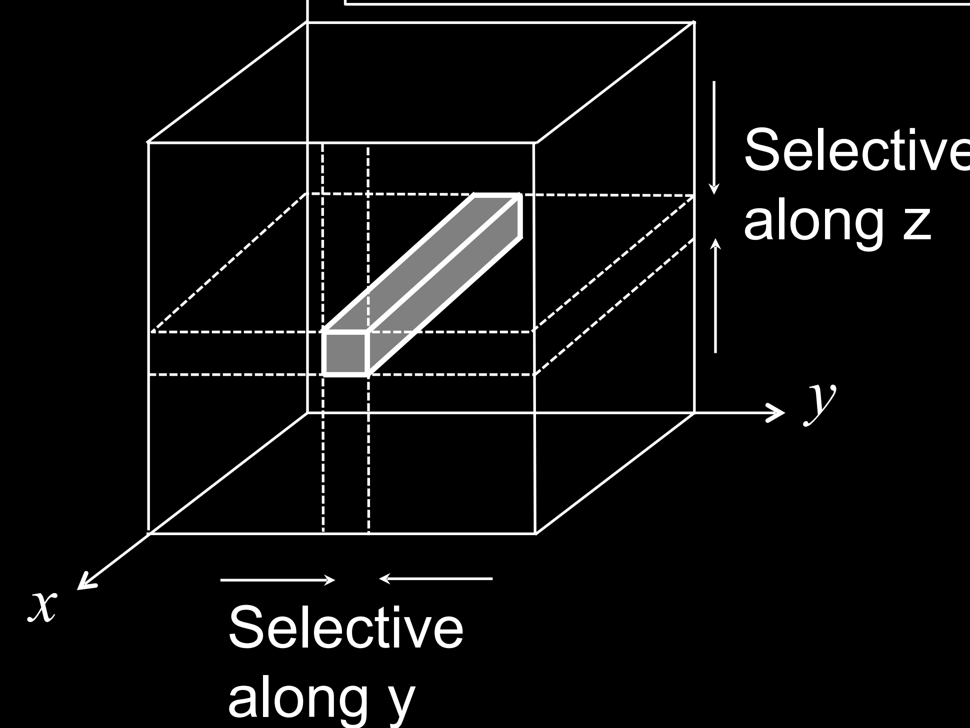
What is Multi-Dimensional Excitation?

Multi-dimensional excitation occurs when using multi-dimensional RF pulses in MRI/NMR, i.e. 2D or 3D RF pulses

1D vs. N-D RF Pulses

2D/N-D Pulse Design Requires:
- Specific B1 waveform
- Specific gradient waveforms



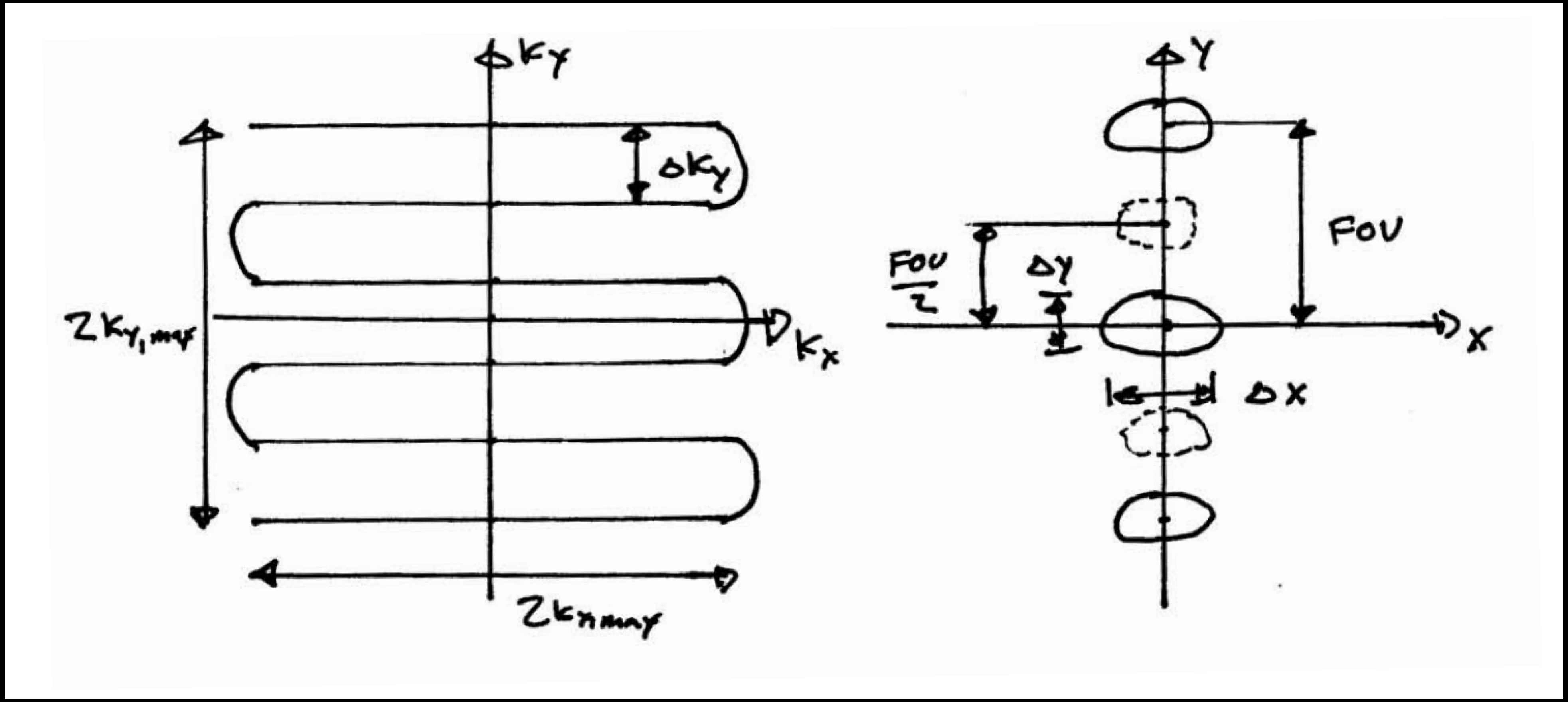


- 1D pulses are selective along 1 dimension, typically the slice select dimension
- 2D pulses are selective along 2 dimensions
  - So, a 2D pulse would select a long cylinder instead of a slice
  - The shape of the cross section depends on the 2D RF pulse

2D EPI Pulse Design

Designing EPI k-space Trajectory

- Ideally, an EPI trajectory scans a 2D raster in k-space



Resolution? / FOV?

Designing EPI k-space Trajectory

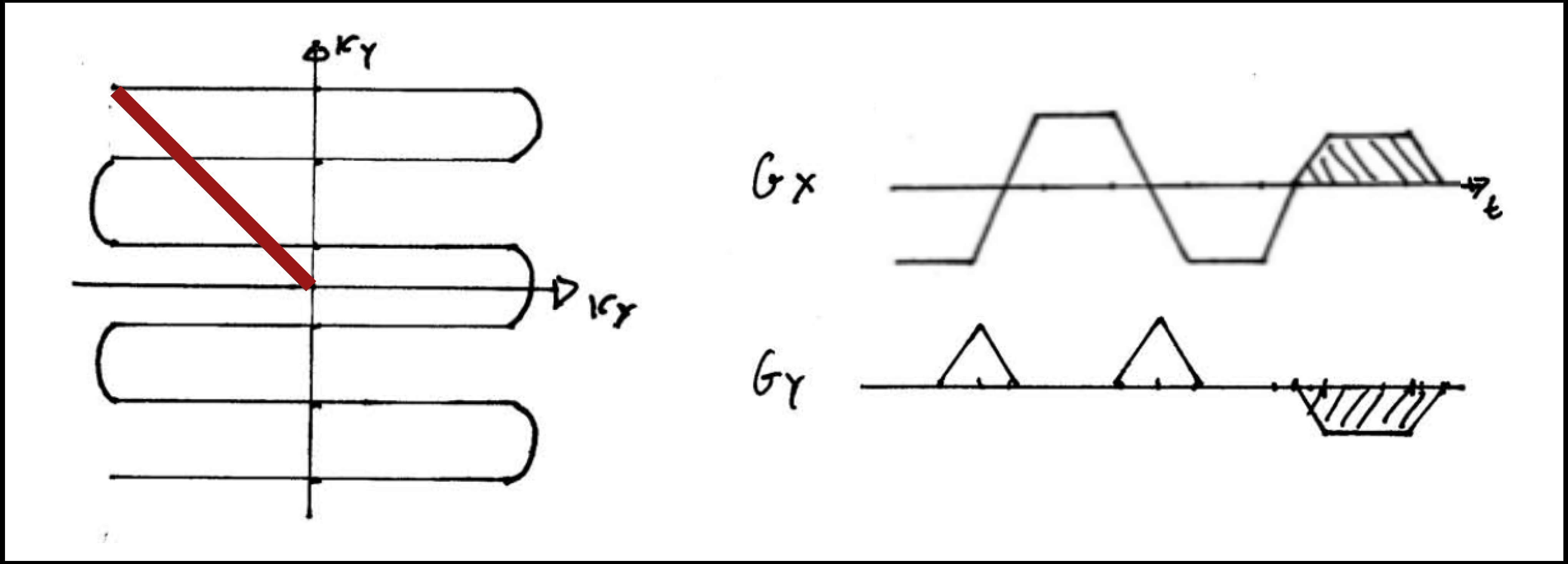
- Resolution: $$\Delta x = \frac{TBW}{2k_{x,max}} \quad \Delta y = \frac{TBW}{2k_{y,max}}$$
- FOV = $1/\Delta k_y$

$$\Delta k_y = \frac{2k_{y,max}}{L-1}$$

- Ghost FOV = FOV/2
  - Eddy currents & delays produce this

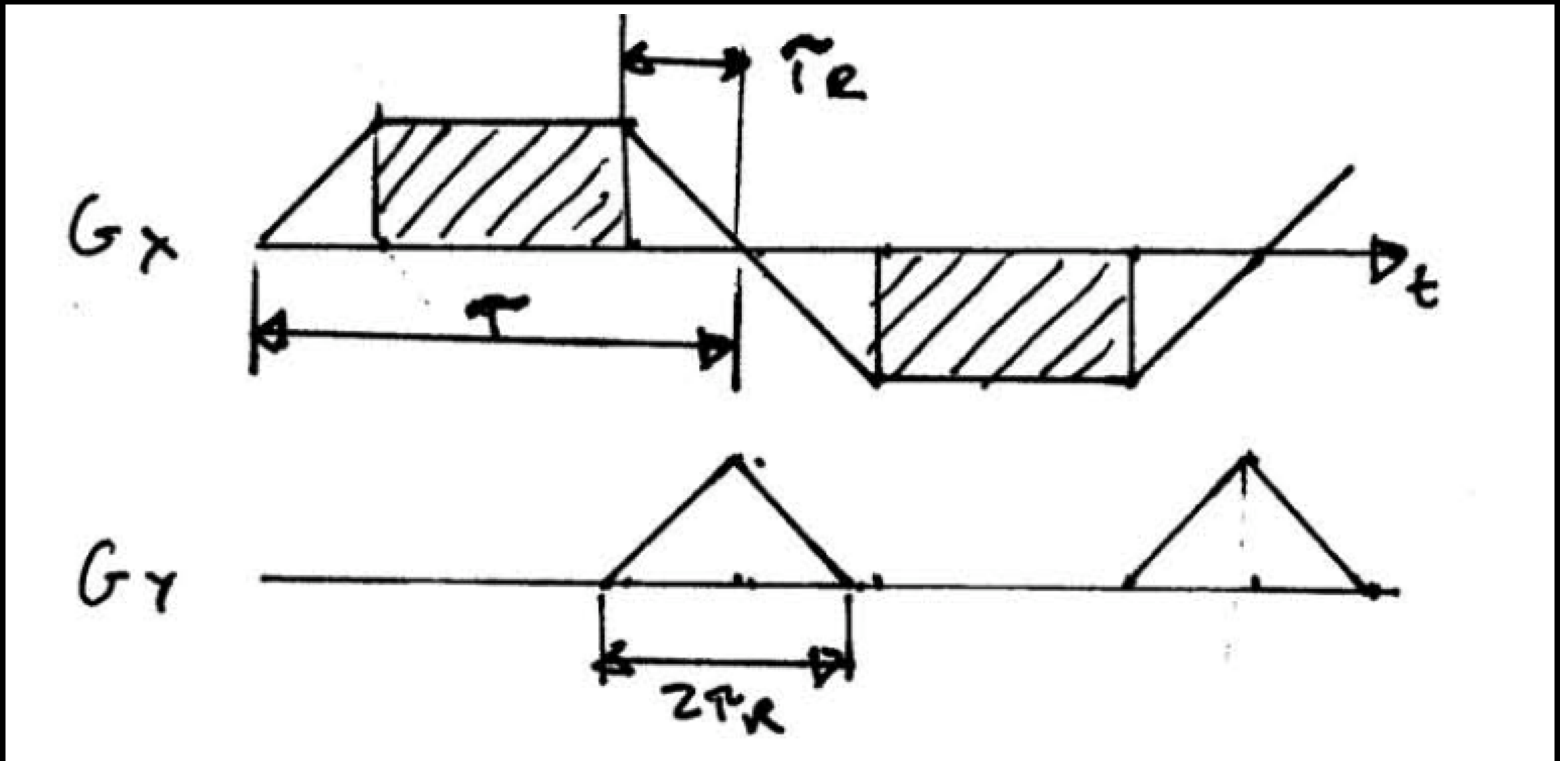
Designing EPI k-space Trajectory

- Refocusing gradients
  - Returns to origin at the end of pulse
  - (Consider trajectory in excitation k-space)



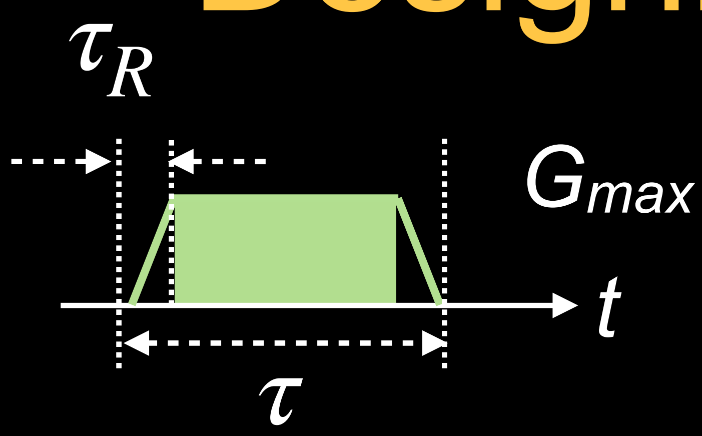
Designing EPI Gradients

- Designing readout lobes and blips
  - Flat-top only design



  - RF only played during flat part (simpler)

Designing EPI Gradients: G_x



$$\tau = 1ms \qquad \tau_R = 1/4ms$$

$$2k_{x,max} = \frac{\gamma}{2\pi}(\tau - 2\tau_R) \cdot G_{max}$$

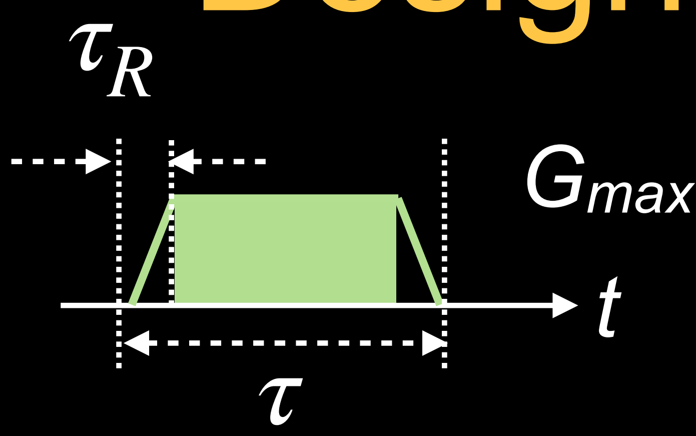
$$= 4.257[kHz/G] \cdot \frac{1}{2}[ms] \cdot 4[G/cm]$$

$$= 8.514[cycles/cm]$$

$$\Delta x = \frac{TBW}{2k_{x.max}}$$

$$TBW = 1 : \Delta x = \frac{1}{8.514[cycles/cm]} \approx 0.12[cm]$$

Designing EPI Gradients: G_x



$$\tau = 1ms \qquad \tau_R = 1/4ms$$

$$2k_{x,max} = \frac{\gamma}{2\pi}(\tau - 2\tau_R) \cdot G_{max}$$

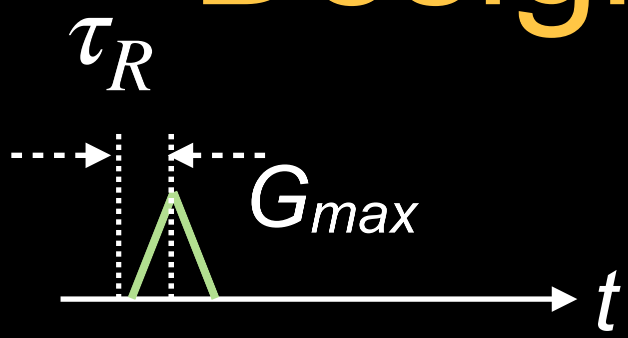
$$= 4.257[kHz/G] \cdot \frac{1}{2}[ms] \cdot 4[G/cm]$$

$$= 8.514[cycles/cm]$$

$$\Delta x = \frac{TBW}{2k_{x.max}}$$

$TBW = 4 : \Delta x \approx 0.47[cm]$ *(More typical)*

Designing EPI Gradients: G_y



$$\tau_R = 1/4ms$$

$$\Delta k_y = \frac{\gamma}{2\pi} \cdot \frac{1}{2} \cdot 2\tau_R \cdot G_{max}$$

$$= 4.257[kHz/G] \cdot \frac{1}{4}[ms] \cdot 4[G/cm] \quad = 4.257[cycles/cm]$$

Assume L = 11 (k-space lines)

$$2k_{y,max} = (L-1) \cdot \Delta k_y = 42[cycles/cm]$$

$$\Delta y = \frac{TBW = 1}{2k_{y,max}} = 0.024[cm] \qquad \mathrm{FOV} = \frac{1}{\Delta k_y} = 0.23[cm]$$

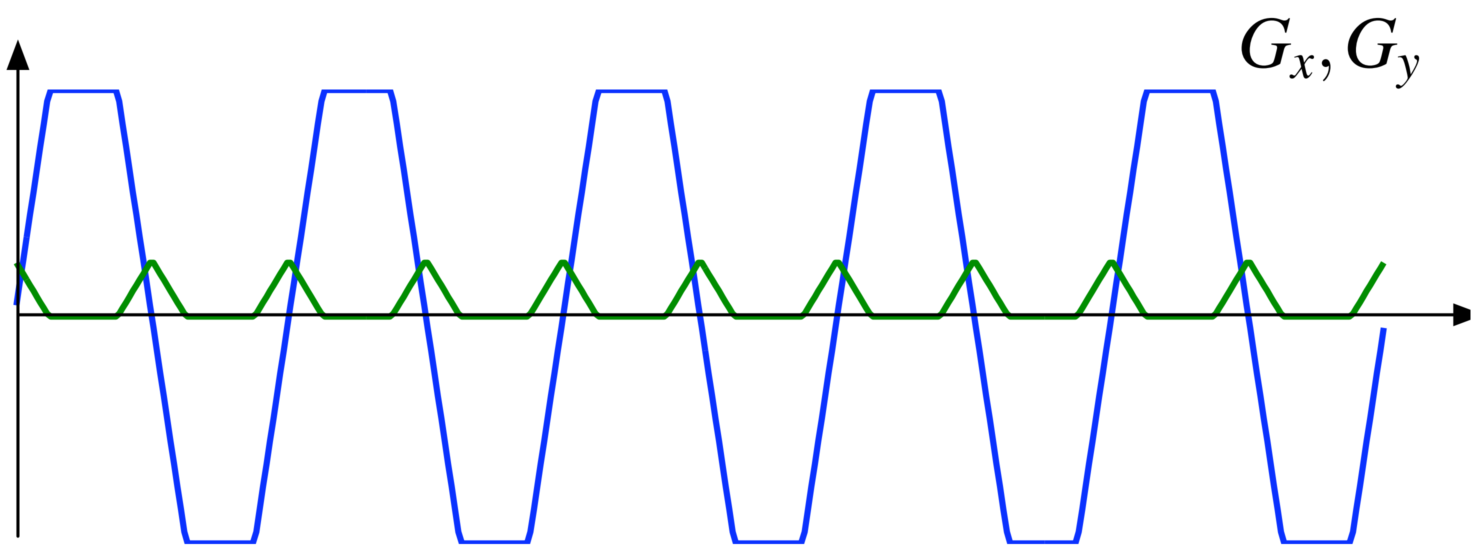
Designing EPI Gradients

- Easy to get k-space coverage in k_y
- Hard to get k-space coverage in k_x
- We can get more k-space coverage by
  - making blips narrower
  - playing RF during part of ramps

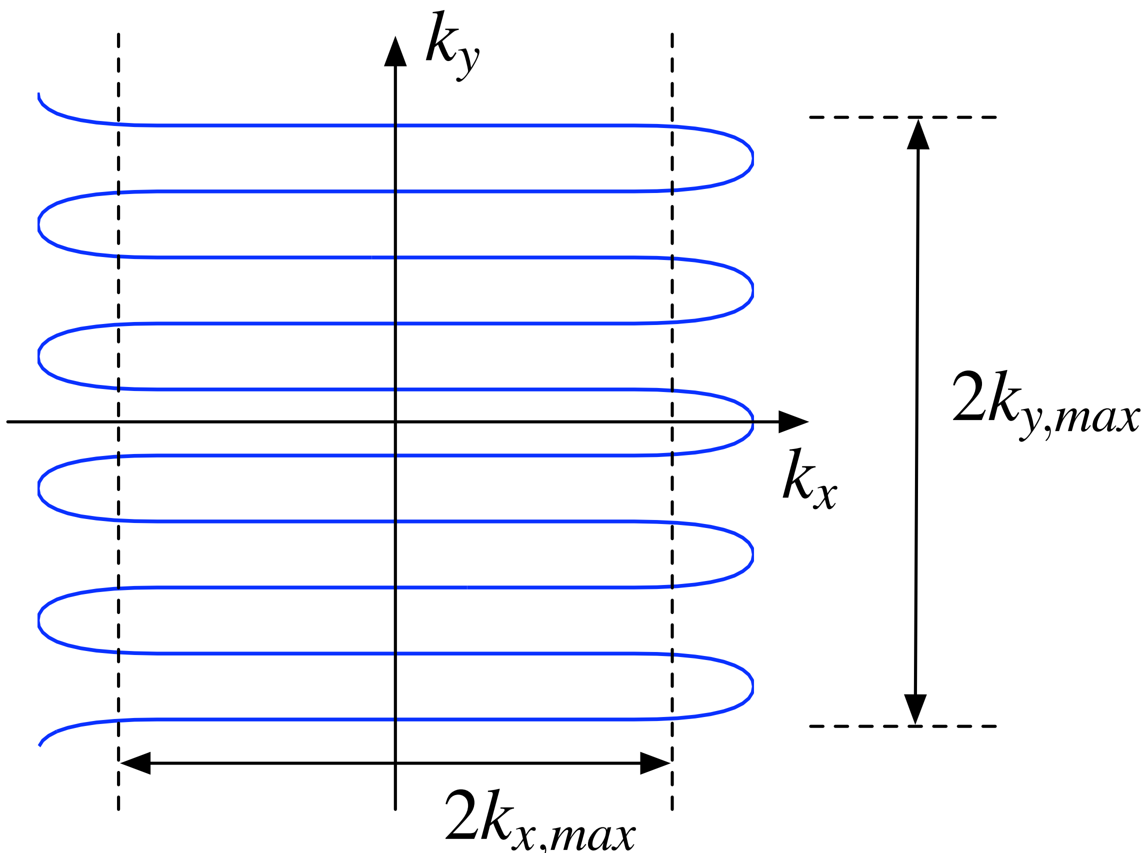
Blipped EPI

- Rectilinear scan of k-space
- Most efficient EPI trajectory
- Common choice for spatial pulses
- Sensitive to eddy currents and gradient delays

Blipped EPI



Gradient Waveforms

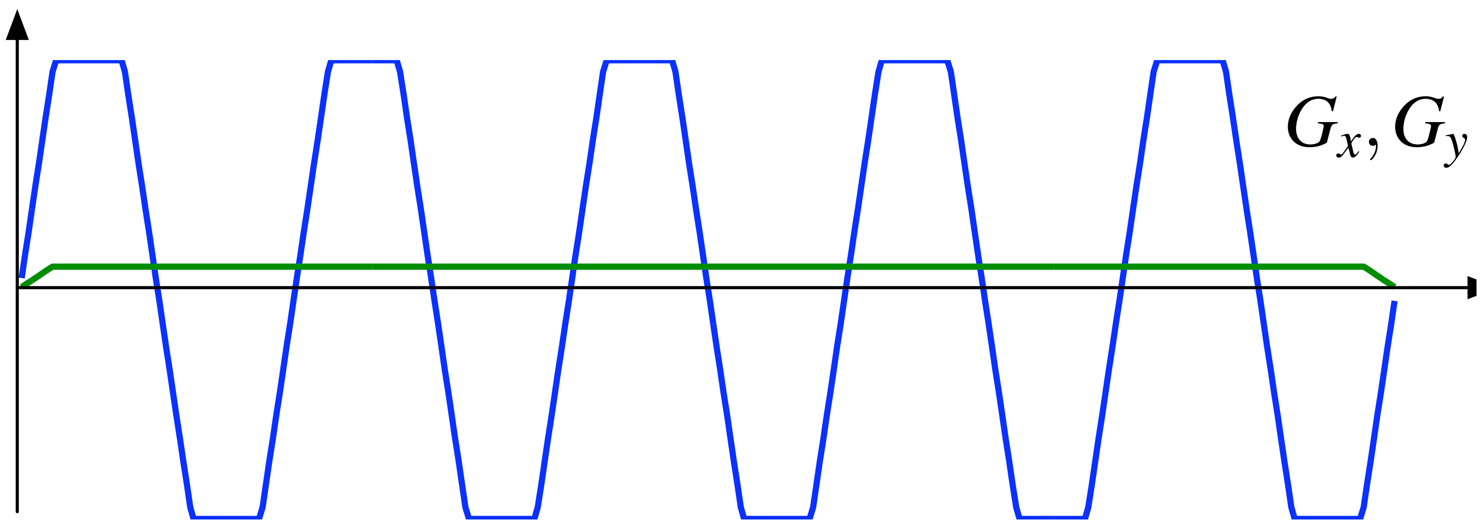


k-Space Trajectory

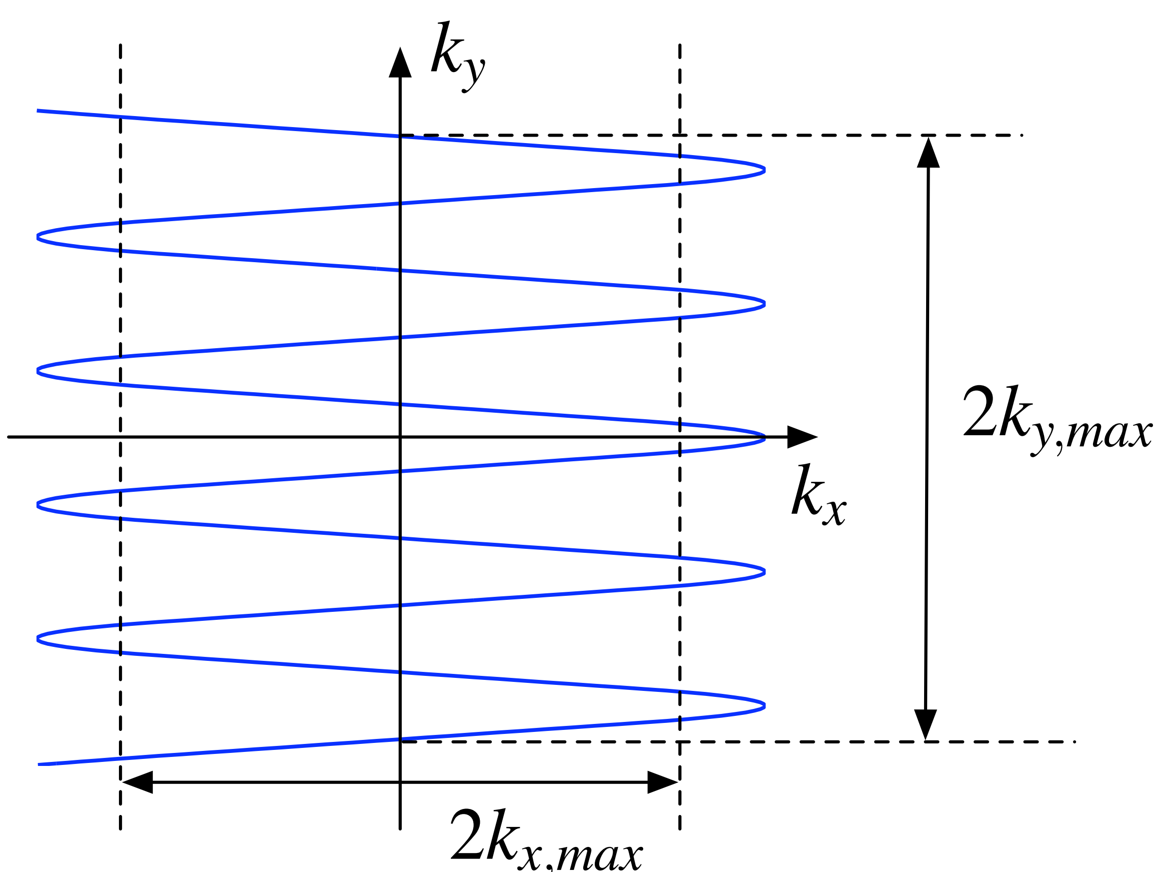
Continuous EPI

- Non-uniform k-space coverage
- Need to oversample to avoid side lobes
  - Less efficient than blipped
- Sensitive to eddy currents and gradient delays
  - Only choice for spectral-spatial pulses

Continuous EPI



Gradient Waveforms

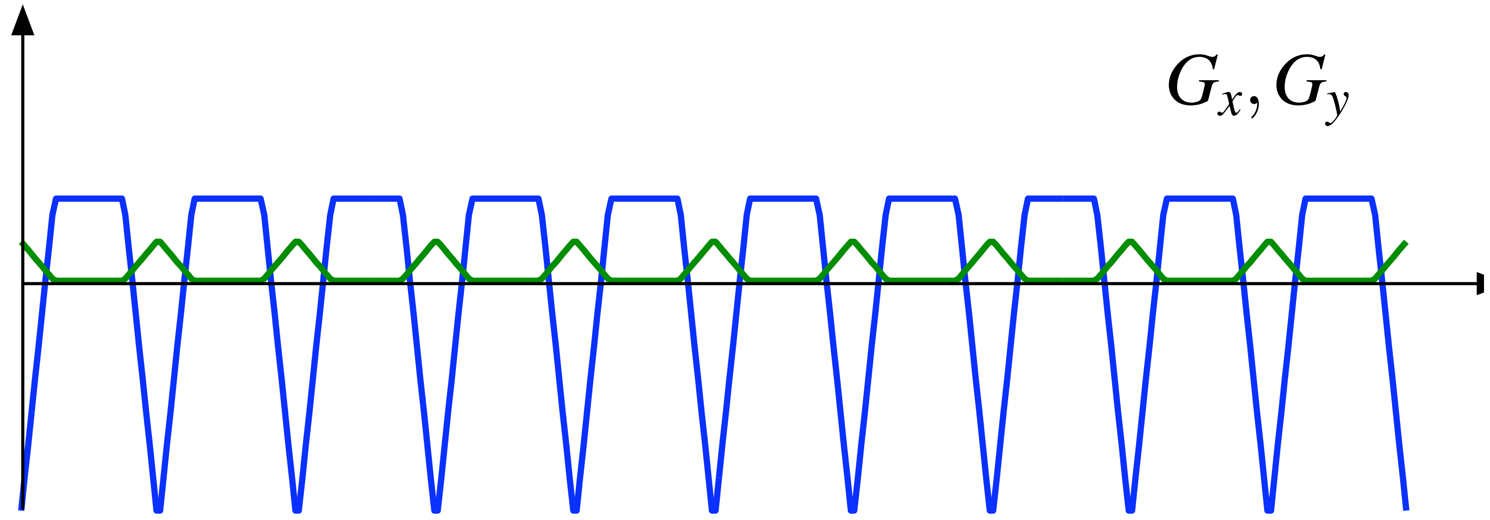


k-Space Trajectory

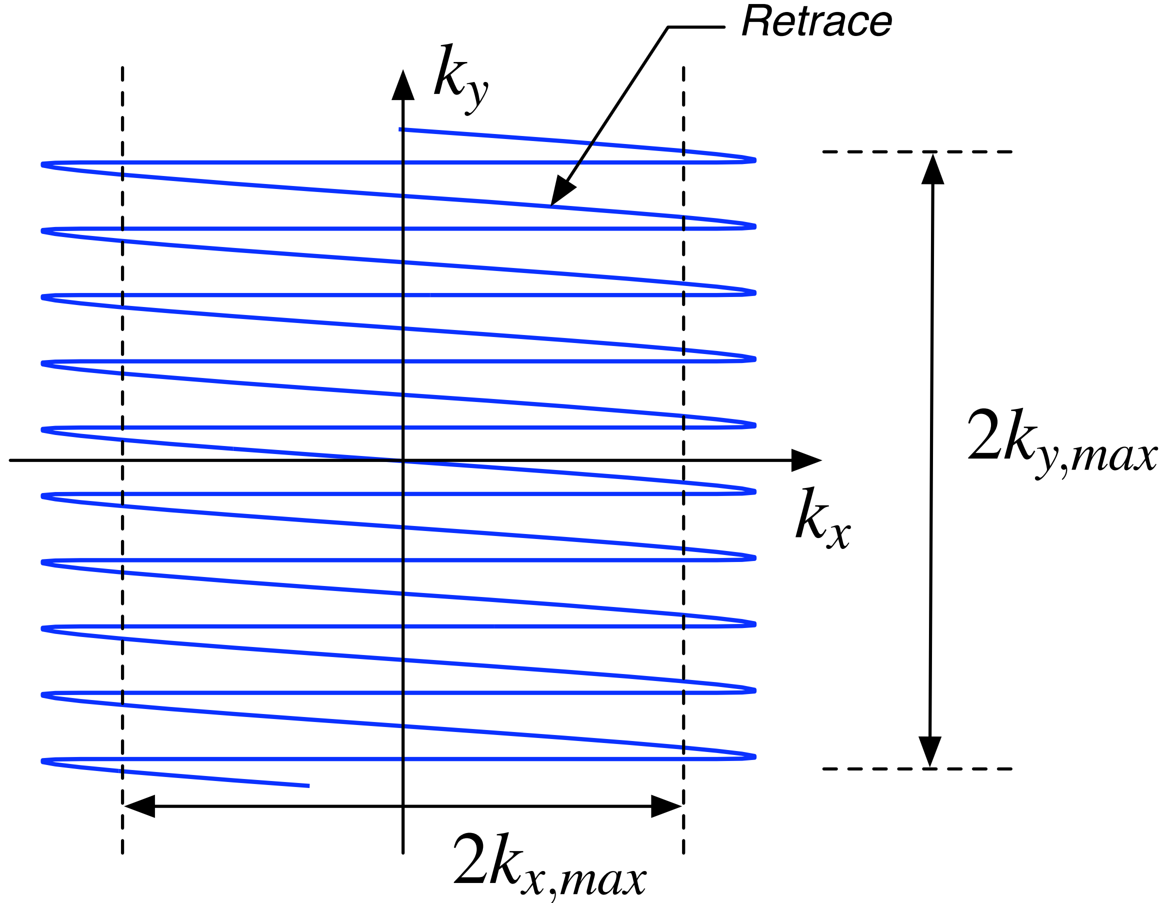
Flyback EPI

- Can be blipped or continuous
- Less efficient since retraces not used (depends on gradient system)
- Almost completely immune to eddy currents and gradient delays

Flyback EPI



Gradient Waveforms



k-Space Trajectory

Designing 2D EPI Spatial Pulses

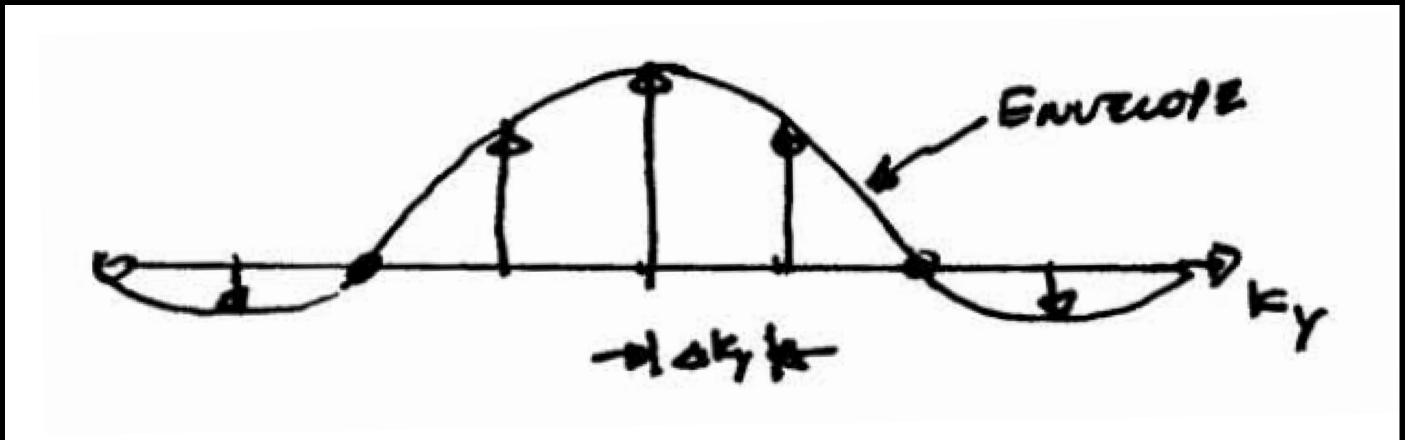
- Two major options
  - General approach, same as 2D spiral pulses
  - Separable, product design (easier)
- General approach
  - Choose EPI k-space trajectory
  - Design gradient waveforms
  - Design *W(k)*, k-space weighting
  - Design $B_1(t)$

Separable, Product Design

- Assume,

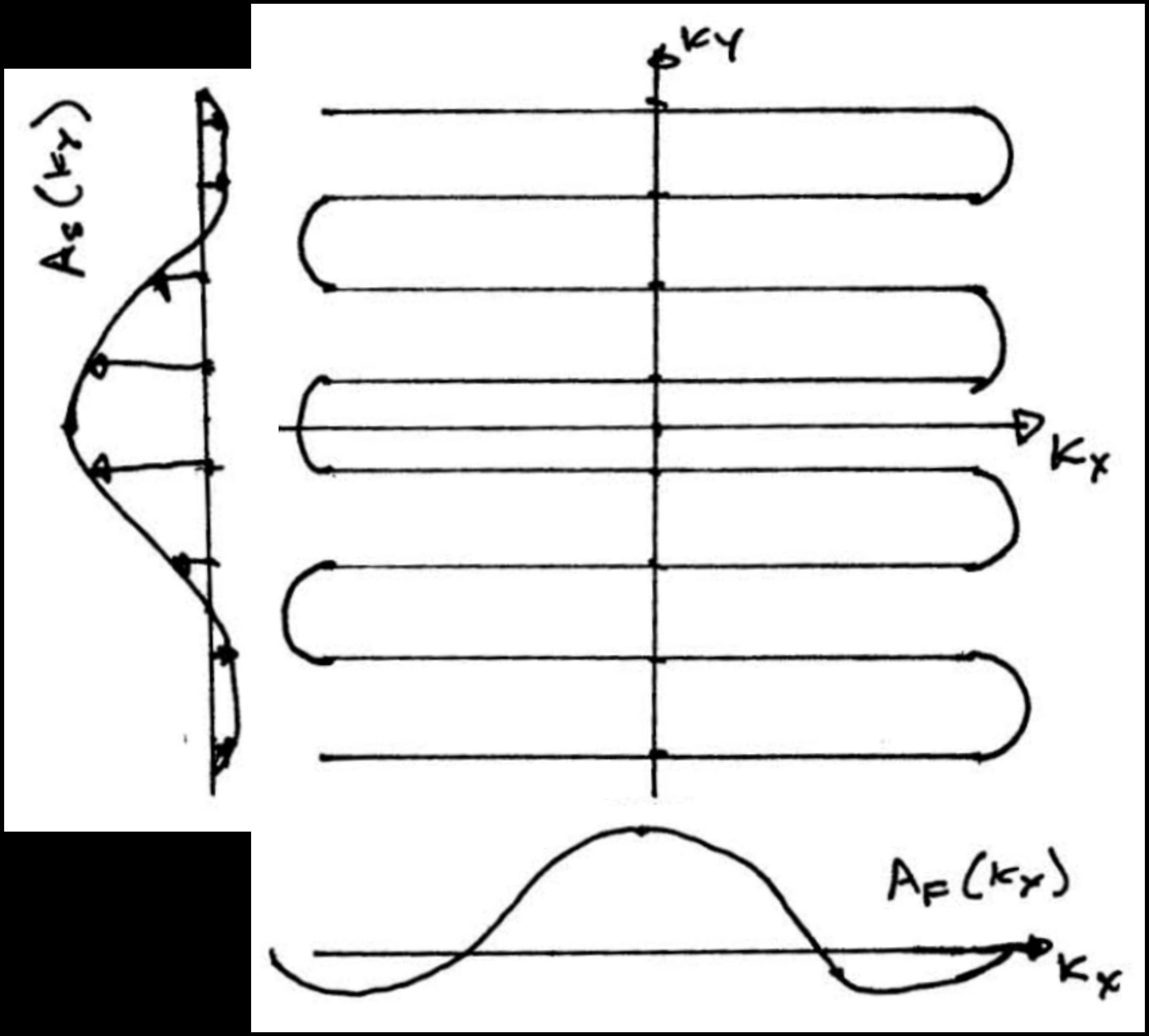
$$W(k_x, k_y) = A_F(k_x) \cdot A_S(k_y)$$

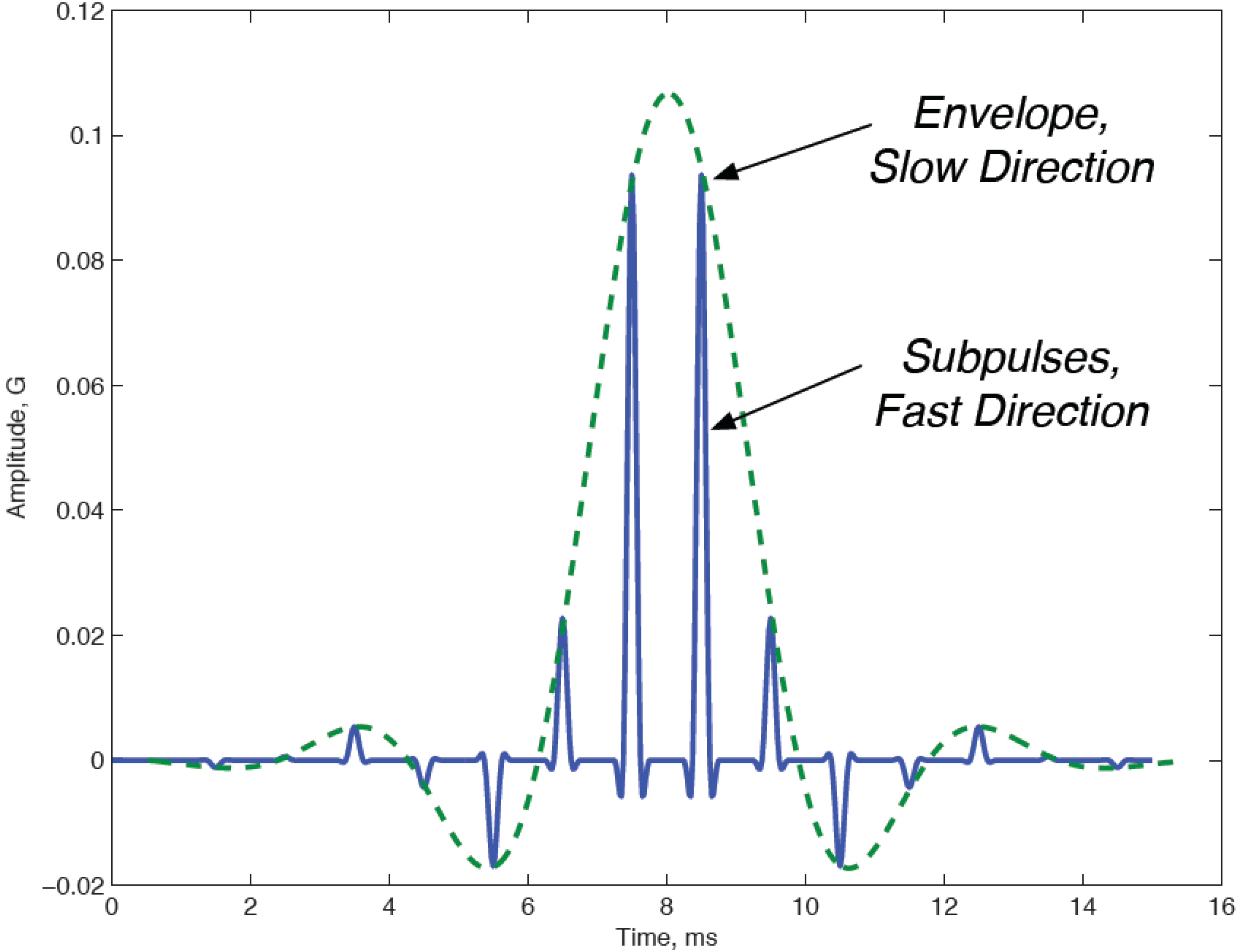
$A_S(k_y)$: weighting in the slow, blipped direction
$A_F(k_x)$: weighting in the fast oscillating direction



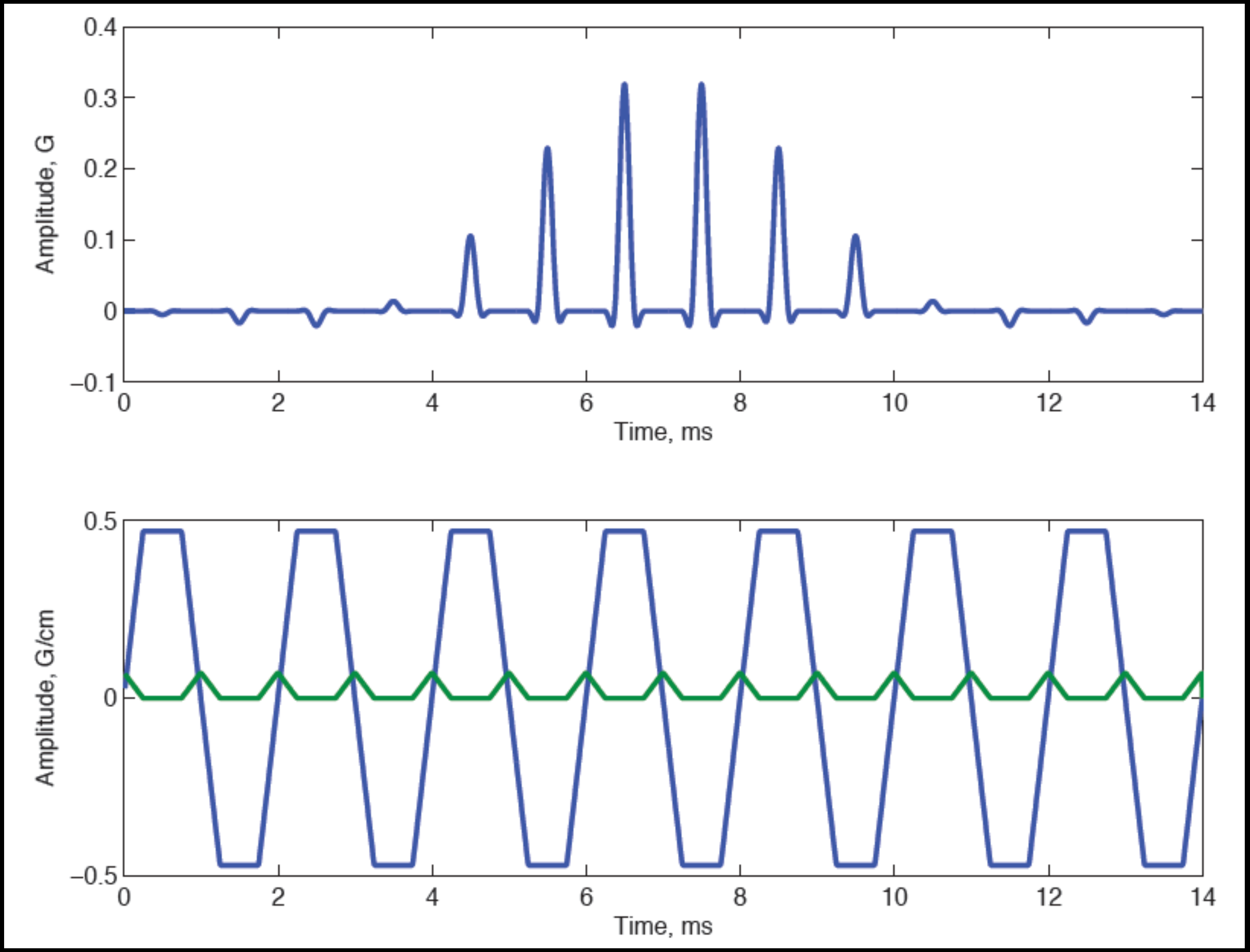
- Each impulse corresponds to a pulse in the fast direction, *$A_F(k_x)$*

Separable, Product Design

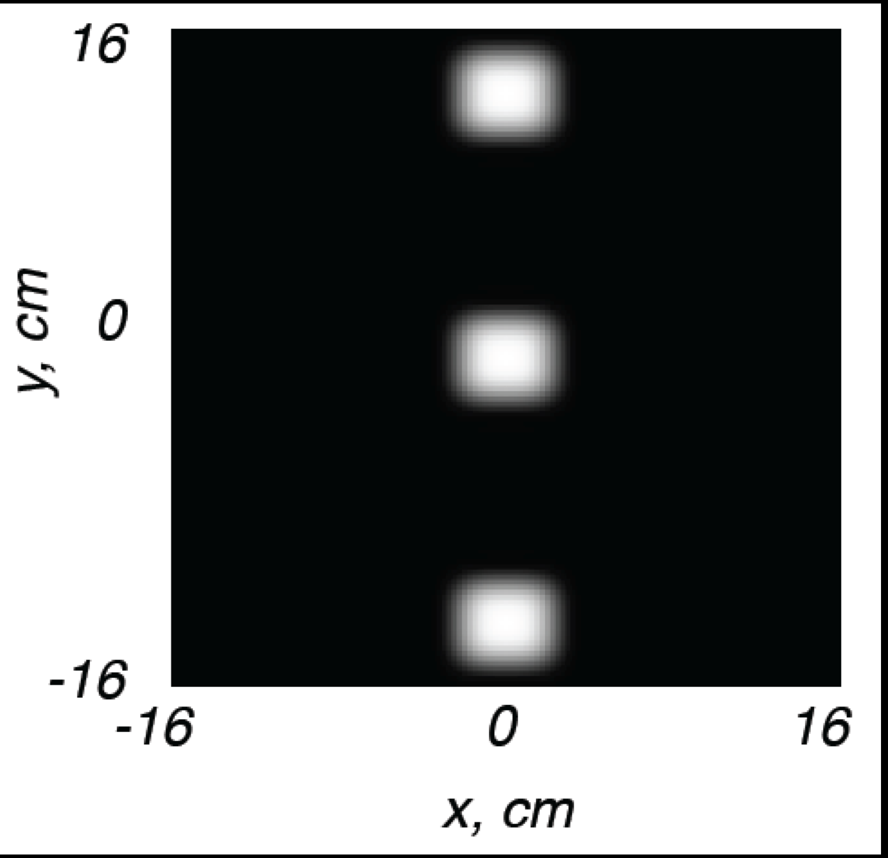




Envelope,
Slow Direction
Subpulses,
Fast Direction
Amplitude, G
Time, ms
0.12
0.1
0.08
0.06
0.04
0.02
0
−0.02
0
2
4
6
8
10
12
14
16



1 ms subpulses
14 subpulses
Flattop only (0.5 ms)
4 cm x 4 cm mainlobe
Sidelobes at +/- 13 cm



MATLAB Demo

Bloch Simulator

- Code from Dr. Brian Hargreaves

```
[mx,my,mz] = bloch(b1,gr,tp,t1,t2,df,dp,mode,mx,my,mz)

Bloch simulation of rotations due to B1, gradient and
off-resonance, including relaxation effects.  At each time
point, the rotation matrix and decay matrix are calculated.
Simulation can simulate the steady-state if the sequence
is applied repeatedly, or the magnetization starting at m0.

INPUT:
        b1 = (Mx1) RF pulse in G.  Can be complex.
        gr = (Mx1,2,or 3) 1,2 or 3-dimensional gradient in G/cm.
        tp = (Mx1) time duration of each b1 and gr point, in seconds,
                        or 1x1 time step if constant for all points
                        or monotonically INCREASING endtime of each
                        interval..
        t1 = T1 relaxation time in seconds.
        t2 = T2 relaxation time in seconds.
        df = (Nx1) Array of off-resonance frequencies (Hz)
        dp = (Px1,2,or 3) Array of spatial positions (cm).
                Width should match width of gr.
        mode= Bitmask mode:
                Bit 0:  0-Simulate from start or M0, 1-Steady State
                Bit 1:  1-Record m at time points.  0-just end time.
```

Windowed Sinc RF Pulse

```
%% Design of Windowed Sinc RF Pulses
tbw = 4;
samples = 512;

rf = wsinc(tbw, samples);
```

```
function h = wsinc(tbw, ns)

% rf = wsinc(tbw, ns)
%
%   tbw  --  time bandwidth product
%   ns    --  number of samples
%   h   -- windowed sinc function, normalized so that sum(h) = 1

xm = (ns-1)/2;
x = [-xm:xm]/xm;
h = sinc(x*tbw/2).*(0.54+0.46*cos(pi*x));

h = h/sum(h);
```

RF Pulse Scaling

```
%% Plot RF Amplitude
rf = (pi/2)*wsinc(tbw,samples);

pulseduration = 1; %ms
rfs = rfscaleg(rf, pulseduration); % Scaled to Gauss
```

$$\theta = \int_0^\tau \gamma B_1(s)ds$$

$$\theta_i = \gamma B_1(t_i)\Delta t$$

$$B_1(t_i) = \frac{1}{\gamma \Delta t}\theta_i$$

RF Pulse Scaling

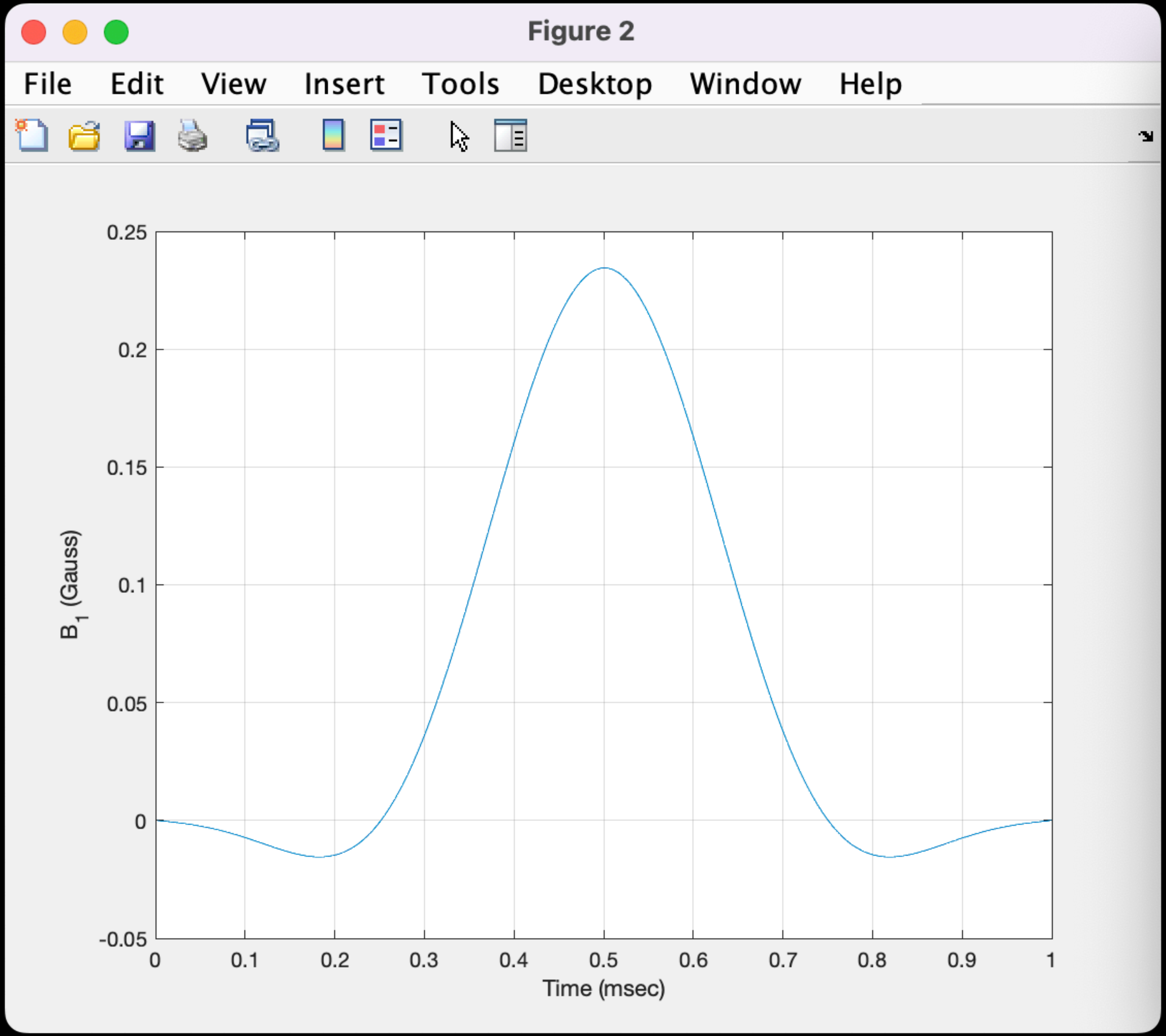
```
%% Plot RF Amplitude
rf = (pi/2)*wsinc(tbw,samples);

pulseduration = 1; %ms
rfs = rfscaleg(rf, pulseduration); % Scaled to Gauss
```

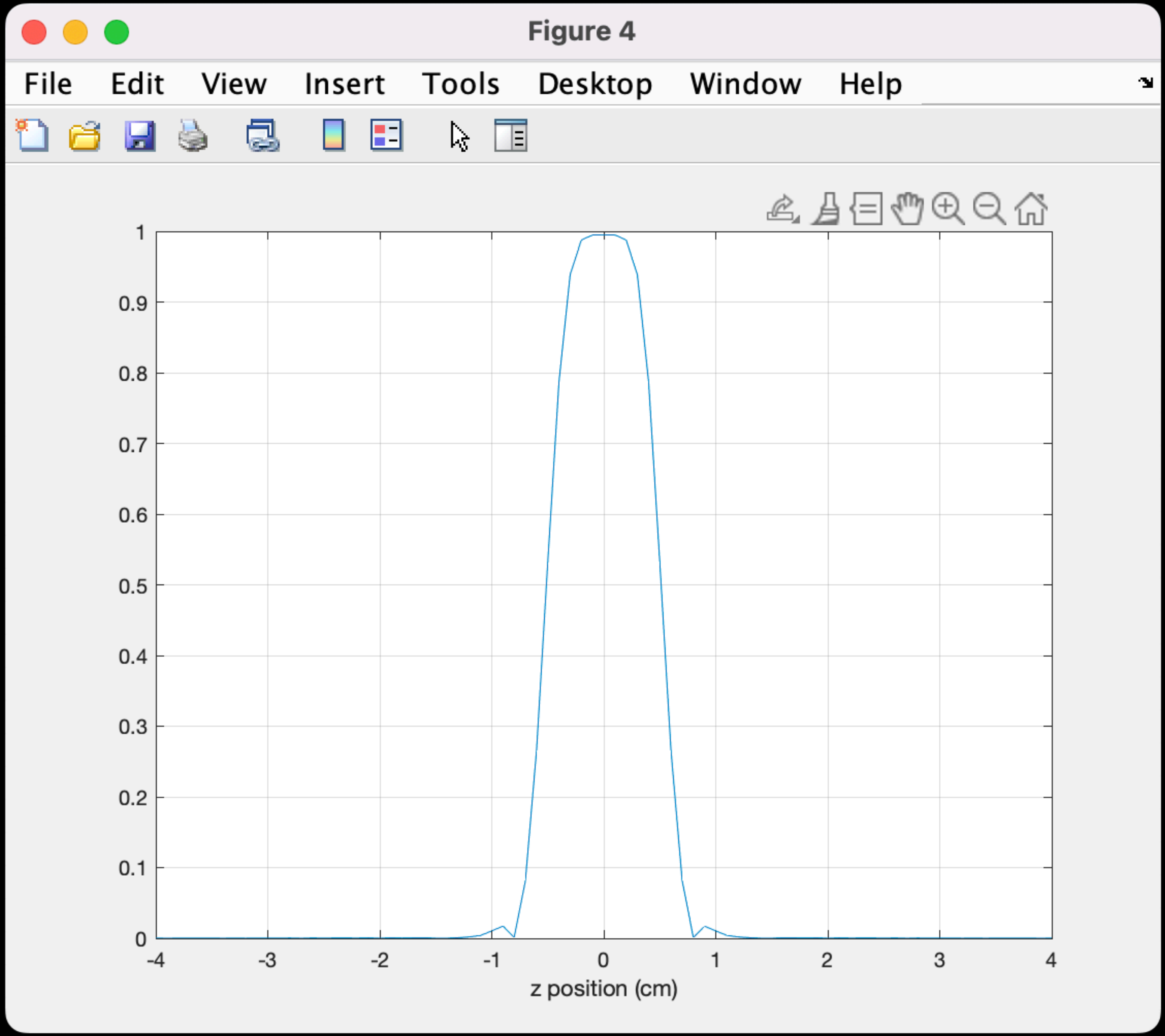
```
function rfs = rfscaleg(rf,t);

%  rfs = rfscaleg(rf,t)
%
%     rf   -- rf waveform, scaled so sum(rf) = flip angle
%     t    -- duration of RF pulse in ms
%     rfs -- rf waveform scaled to Gauss
%

gamma = 2*pi*4.257; % kHz*rad/G
dt = t/length(rf);
rfs = rf/(gamma*dt);
```

Bloch Simulation

```
%% Simulate Slice Profile
tbw = 4;
samples = 512;

rf = (pi/2)*wsinc(tbw,samples);
pulseduration = 1; %ms

rfs = rfscaleg(rf, pulseduration);              % Scaled to Gauss
b1 = [rfs zeros(1,samples/2)];                  % in Gauss
g = [ones(1,samples) -ones(1,samples/2)];       % in G/cm

x = (-4:.1:4);                % in cm
f = (-250:5:250);             % in Hz
dt = pulseduration/samples/1e3;
t = (1:length(b1))*dt;        % in usec

% Bloch Simulation
[mx,my,mz] = bloch(b1,g,t,1,.2,f,x,0);
mxy=mx+1i*my;
```

Bloch Simulation





Thank You!

- Further reading
  - Read “Spatial-Spectral Pulses” p.153-163

- Acknowledgments
  - John Pauly’s EE469B (RF Pulse Design for MRI)
  - Shams Rashid
  - Kyung Sung

Holden H. Wu, PhD
HoldenWu@mednet.ucla.edu
https://mrrl.ucla.edu/wulab